Ambient Spline Approximation on Manifolds

Ambiente Spline-Approximation auf Mannigfaltigkeiten
Zur Erlangung des Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Dissertation von M.Sc. Sonja Odathuparambil aus Frankfurt a.M.
Tag der Einreichung: 20. Mai 2016, Tag der Prüfung: 13. Juli 2016
August 2016 — Darmstadt — D 17

1. Gutachten: Prof. Dr. Ulrich Reif
2. Gutachten: Prof. Dr. Oleg Davydov

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Mathematik
Geometrie und Approximation

Ambient Spline Approximation on Manifolds
Ambiente Spline-Approximation auf Mannigfaltigkeiten

Genehmigte Dissertation von M.Sc. Sonja Odathuparambil aus Frankfurt a.M.

1. Gutachten: Prof. Dr. Ulrich Reif
2. Gutachten: Prof. Dr. Oleg Davydov

Tag der Einreichung: 20. Mai 2016
Tag der Prüfung: 13. Juli 2016

Darmstadt 2016 — D 17

Acknowledgements

First of all, I am grateful to God for the opportunity to work on this topic and the good health and wellbeing that were necessary to complete this thesis.

I would like to express my sincere gratitude to Prof. Dr. Ulrich Reif for his continuous support, for his patience, confidence and motivation. His expert guidance and professional advice helped me during all the time of my research. I thank Prof. Dr. Oleg Davydov who agreed to be the coreferee of this work.

Moreover, I would like to give special thanks to Dr. Nicole Lehmann for a lot of stimulating discussions. I thank Tobias Ewald and Lars-Benjamin Maier for proofreading the thesis and sharing thoughts and ideas on this topic. I also like to thank Dr. Jennifer Prasiswa, who handed over her implementation in MATLAB to me. I made use of her implementiation in some areas with small modifications. I would like to thank all my collegues of the group *Geometry and Approximation*, especially my officemate Dr. Susanne Kürsten, for all the fun we had in the last five years.

Last but not least I thank my family for supporting me.

Contents

Abstract

Approximation in $\mathbb{R}^d$ is already well studied while approximation on manifolds still leads to difficulties. The present thesis introduces three approximation methods on compact manifolds. The manifold is considered as a submanifold embedded in $\mathbb{R}^d$ and the problem is extended to some ambient domain of the submanifold.

The first method returns C^k-approximations, $k \in \mathbb{N}$, of given functions on smooth compact submanifolds. We prove that the method shows optimal approximation behaviour for submanifolds of codimension one. The second and the third method approximate solutions of linear intrinsic PDEs. After extending the problem to some domain around the submanifold boundary conditions are added. The problem is solved by using the Finite Element Method. Numerical test confirm the ideas of the methods.

Deutsche Zusammenfassung

Das Lösen von Partiellen Differentialgleichungen (PDGen) auf Mannigfaltigkeiten ist eine Herausforderung, welche auf unterschiedlichen Bereichen Anwendung findet. In der Physik, Geophysik und Strömungslehre wie auch in der Biologie und Medizin und zunehmend in der Bildverarbeitung trifft man häufig auf solche Probleme. Auch die glatte Approximation gegebener Funktionswerte auf beliebigen Mannigfaltigkeiten hat viele Anwendungsgebiete.

In den vergangenen Jahren gab es mehrfach Ansätze die besagten mathematischen Probleme zu lösen. Dabei haben sich zwei Methoden besonders herausgehoben: Einbettungsmethoden und die Diskretisierung der Fläche. Bei beiden Methoden werden weder Parametrisierungen benutzt, noch müssen irgendwelche komplizierten Funktionenräume auf den Mannigfaltigkeiten konstruiert werden. Die Diskretisierung wird meist bei Fächen in $\mathbb{R}^3$ genutzt. Dabei wird eine Triangulierung der Fäche konstruiert. Dann wird z.B. mit Hilfe der Finite Elemente Methode die Lösung einer PDG approximiert. Diese sind meist linear. Siehe z.B. [DE13] für eine detaillierte Beschreibung. Da viele Eigenschaften, wie z.B. die Krümmung, bei triangulierten Flächen noch nicht gut verstanden sind, stößt man bei diesen Vorgehen häufig auf Hindernisse. Dagegen haben Einbettungsmethoden in den letzten Jahren an Popularität gewonnen. Bei diesen Methoden betrachtet man die Mannigfaltigkeit als eine Untermannigfaltigkeit, die in einen höherdimensionalen Raum eingebettet ist. Das gegebene Problem wird dann in den umgebenden Raum erweitern und dort mittels bekannter Techniken gelöst.

In dieser Arbeit werden drei Einbettungsmethoden vorgestellt. Die erste Methode dient der Funktionenapproximation auf Mannigfaltigkeiten, während die anderen beiden Methoden intrinsische PDGen approximativ lösen. Die *Ambient B-spline Method* konstruiert C^k-Approximationen zu gegebenen Funktionen oder diskreten Funktionswerten auf Untermannigfaltigkeiten. Zunächst wird ein uniformes Gitter über die Untermannigfaltigkeit gelegt. Alle Gitterzellen, welche die Untermannigfaltigkeit schneiden, werden mit Datenpunkten gefüllt. Diese werden anschließend mit Tensorprodukt B-splines approximiert. Der so erhaltene Spline wird dann auf die Untermannigfaltigkeit eingeschränkt. Die *Ambient B-spline Method* kann auf allen glatten kompakten Untermannigfaltigkeiten unabhängig von der Dimension, der Kodimension oder dem Geschlecht angewendet werden und liefert beliebig glatte Approximationen. Sie ist unkompliziert, benötigt keine Parametrisierung oder Triangulierung der Mannigfaltigkeit, profitiert von den positiven Eigenschaften der B-splines und ist leicht zu implementieren. Außerdem liefert sie für Untermannigfaltigkeiten von Kodimension eins optimale Approximationsgüte. Das bedeutet, der Fehler der Approximation konvergiert in $\mathcal{O}(h^n)$, wobei $h > 0$ die Gitterweite und $n \in \mathbb{N}$ die Ordnung der B-splines ist. Der Beweis der Konvergenzordnung ist ein Schwerpunkt dieser Arbeit. Der Beweis, wie er in dieser Arbeit dargestellt wird, lässt sich nicht auf Untermannigfaltigkeiten von höherer Kodimension übertragen. Mehrere numerische Beispiele für 2-Mannigfaltigkeiten im $\mathbb{R}^3$ bestätigen die Theorie.

Die zweite Methode, die *Ambient Signed Distance Method*, löst lineare elliptische PDGen zweiter Ordnung gegeben auf Untermannigfaltigkeiten. Die gegebene PDG wird in den umgebenden Raum übersetzt, sodass die Lösungen beider PDGen auf der Untermannigfaltigkeit übereinstimmen. Dabei werden die Differentialoperatoren mit Hilfe der Abstandsfunktion modifiziert. Randbedingungen werden in einem neuem Sinne definiert und das System wird mit der Finiten Elemente Methode gelöst. Falls die intrinsische PDG lösbar ist, so liefert die *Ambient Signed Distance Method* eine beliebig glatte Approximation ihrer Lösung. Sowohl der Workflow als auch die Implemen-

tierung der Methode sind leicht verständlich und unkompliziert. Numerische Experimente zeigen sehr gute Ergebnisse, die auf einer Implementierung in MATLAB basieren.

Die *Ambient Level Set Method* funktioniert ähnlich wie die zuvor beschriebene *Ambient Signed Distance Method*, jedoch ohne von der Abstandsfunktion Gebrauch zu machen. Die Methode löst PDGen auf implizit beschriebenen Untermannigfaltigkeiten. Die Herleitung der Methode bildet einen weiteren Schwerpunkt dieser Arbeit. Die Methode ist unkompliziert und einfach zu implementieren. Eine Implementierung in MATLAB und einige numerische Versuche werden in dieser Arbeit vorgestellt.

Sowohl die *Ambient Signed Distance Method* als auch die *Ambient Level Set Method* können auf allen glatten, kompakten Untermannigfaltigkeiten von Kodimension eins eingesetzt werden und liefern glatte Approximationen von linearen intrinsischen elliptischen PDGen zweiter Ordnung. Die Grundideen dieser Methoden lassen sich gut verallgemeinern und können auch zum Lösen parabolischer PDGen genutzt werden. Desweiteren sind beide Methoden so konzipiert, dass komplizierte Terme vermieden werden und alle Integrationsgebiete mehrdimensionale Intervalle sind. Die Einfachheit und andere positive Eigenschaften der Tensorprodukt B-splines werden genutzt und gleichzeitig wird den Stabilitätsproblemen von Tensorprodukt B-splines auf beliebigen Gebieten vorgebeugt. Ein enormer Vorteil beider Methoden ist, dass keine Projektion genutzt wird. Dies unterscheidet die Methoden von ähnlichen Ansätzen, siehe z.B. [BCOS01] oder [DE13]. In den dort beschriebenen Methoden geht Elliptizität auf Grund der Projektion verloren. Eine theoretische Analyse der Konvergenzordnung und eine Verallgemeinerung auf andere PDGen bleiben offen.

1 Introduction

One important problem in medical imaging is to find feature points or curves on the cortical surface of the brain. These curves are important information to neuroscientists who study brain diseases. Here the problem can be traced back to a mathematical problem, namely intrinsically solving partial differential equations for data defined on manifolds with arbitrary geometry. This is a remaining challenge within the subject of numerical analysis of partial differential equations (PDEs). Especially surface partial differential equations arise in a variety of applications: Traditionally they appear in fluid dynamics and material science and more recently in the mathematics of images. They also find applications in biology, geometry and of course in computer graphics.

The closely related problem of finding approximating functions on manifolds presents a very interesting area with lots of application, too. This can be seen as a problem itself or as a subproblem of the before mentioned intrinsical PDEs. In a number of applications, functions on arbitrary topology need to be approximated by smooth functions. Mathematical physics, fluid dynamics, image processing, medical imaging, computer graphics and pattern formation are examples of fields of application, and the list becomes longer.

Therefore it is not surprising that there exist many approaches to handle these topics. Many related works define functions on surfaces using parametrisations. For these create a C^k-atlas and define basis functions on each chart. The functions on the manifold are then defined with the help of compositions of these basis functions and the parametrisation.

Using surface parametrisation does not turn out to be the most beneficial way. parametrisations are often difficult to derive for complicated surfaces. Especially the generation of parametrisations of arbitrary differentiability requires elaborate work. Furthermore, the need of patching to connect different parametrisation neighbourhoods is a disadvantage. Depending on the topology of the manifold extraordinary points appear. Except in the case of tori the existence of extraordinary points is unavoidable on closed surfaces. Finally, if a function is a polynomial on one chart it does not have to be polynomial on a different overlapping chart.

The last mentioned drawback was overcome by an alternative idea: A new functional space is created on the manifold itself. The so called *manifold splines* were introduced by Gu, He and Qin in [GHQ06] in 2006. They construct splines whose domain are manifolds of arbitrary geometry with or without boundary. With manifold splines we can approximate functions on manifolds or solve elliptic PDEs on surfaces. Yet, the construction of the manifold spline space is a difficult task itself. Moreover, the transition functions of the parametrisation need to be affine.

Now, we rewind some existing methods to constitute manifolds or submanifolds without using parametrisation. These are suited better to define functions or even solve PDEs on them. One common alternative is the discretisation of the manifold: The manifold is represented as a union of simplices. Another opportunity is provided by implicit methods: The manifold is described as a submanifold embedded in a higher dimensional space. The advantage of implicit methods is that the complexity associated with equations on manifolds is removed at the expense of solving an equation in a higher dimensional but less complicated space. Obviously these are not the only manifold representations but both these representations are suitable to define functions on manifolds or even solve intrinsic PDEs. See [HG00] for even more representations currently available and their advantages and disadvantages. Let us now gain an insight in how techniques work on simplicial surfaces and embedded submanifolds. The avoidance of charts in the problem formulation as well as in the numerical methods is an advantage of both concepts.

Simplicial Surfaces

One common approach to represent submanifolds is to discretise them by using meshes. This method is mainly used for surfaces. Mesh-based representations are defined by a finite collection of vertices and their connectivity. The continuous surface ω is replaced by a piecewise polynomial surface ω_h. This polynomial surface ω_h lies within a δ-strip around ω such that there exists a bijective correspondence between ω and ω_h.

The most widely spread approach are triangulated surfaces. In that case ω_h consists of finitely many non-degenerate triangles or in case of higher dimension ω_h is the union of finitely many non-degenerate n-simplices. The vertices are taken to sit on the smooth surface ω.

In applications generally the data are discretely defined on the surface. We then need to find a suitable triangulated or polygonal form of the surface. Solving a PDE in this representation involves a nontrivial discretisation of the equations. Furthermore, there occur difficulties when computing other quantities like the gradient or the Laplacians. In [War08] Wardetzky characterises an intrinsic discretisation of the Laplace-Beltrami operator. Mean curvature functionals are also defined in this paper. So, intrinsic PDEs can be discretised and solved. Yet, this approach still leads to numerous difficulties because discretising differential operators is still less well understood than methods on Cartesian grids.

In [DE13] we find another way to discretise differential operators. They define the tangential gradient of a function $g\colon \omega_h \to \mathbb{R}$ on the discrete surface using a projection of some extension G of g

$$\nabla_{\omega_h} g := P_h \cdot \nabla G.$$

The matrix $(P_h)_{ij} = \delta_{ij} - (\nu_h)_i(\nu_h)_j$ with ν_h being the normal on ω_h is the projection on each simplex. A finite element space is set up in order to solve the discretised PDEs. For details on this Surface Finite Element Method (SFEM) see [DE13].

The arising error of this approach depends on the size of the triangles. Dzuik and Elliot have proven that the SFEM leads to an error that converges in $\mathcal{O}(h^2)$ where h is the maximal diameter of the triangles.

Since they provide a flexible representation of manifolds mesh based modelling is commonly used in computer graphics. Yet, as a general technique this approach leads to a number of difficulties as mentioned above. The simplicial surface itself gives limited accuracy which can be overcome by a very dense point cloud. Yet, this costs a lot of memory as well as calculation time. Apart from that, the construction of a simplicial mesh is a non-trivial problem on its own.

Embedding methods

Another approach that attends more and more interest in present works is to represent manifolds as submanifolds embedded in a higher dimensional space. In this implicit representation the submanifold is described as a level set, often the zero-level set, of a higher dimensional function. With this representation there exist different techniques to approximate functions or solve PDEs on the submanifold.

The idea is to formulate a PDE posed on a narrow band around the submanifold such that the solution of the PDE when restricted to the submanifold provides the solution of the original problem. Such a technique was introduces by Bertalmío, Cheng, Osher and Sapiro in 2001 (see [BCOS01]). They solve the PDE in the Cartesian coordinate system by using a projection

$$P_\varphi := \mathbb{1} - \frac{\nabla\varphi \otimes \nabla\varphi}{|\nabla\varphi|^2}$$

where $\nabla\varphi$ is the gradient of the level set function. The embedding PDE is then solved on each level set using classical schemes of forward and backward differences. The method has several

drawbacks: The solution can be discontinuous at the boundary when fixing Dirichlet boundary conditions. Choosing appropriate boundary conditions will not be easy to generalise. Furthermore, reextension is needed so that the solution remains constant in normal direction. In 2003 Greer modified the new framework that was opened up by Bertalmío et. al. (see [Gre03]). Under the condition of φ being the signed-distance function he modified the projection matrix and therewith added some regularity improvements to the method. Both papers concentrate on parabolic differential equations and limit to submanifolds of codimension one.

Another approach was presented by Ruuth and Merriman in [RM08] in 2008. Their embedding method is based on the use of a closest-point-representation on a band around the surface rather than a level set representation. They avoid to involve a projection matrix and construct the embedding PDE as the natural extension of the original. They solve the embedding PDE using standard finite differences on a Cartesian grid in the computational domain combined with a time step method. Since the evaluations cannot agree for long times the embedding PDE has to be reconstructed after each time step. This increases the computational work and causes accumulating errors. Ruuth and Merriman concentrate on parabolic PDEs. This method as well as Greer's method give second order convergence of the error.

Recently, Dziuk and Elliott presented an embedding method to solve elliptic surface PDEs using the Finite Element Method (see [DE13] or [DDEH10]). Just as the method of Bertalmío et. al. the method is based on formulating an embedding PDE with the help of a projection matrix. This embedding PDE is defined on a band around the submanifold where level sets of φ form the boundary. This embedding PDE is then solved by the Finite Element Method. Therefore, linear hat functions are used on a triangulation of the ambient domain. The method converges with the expected order $\mathcal{O}(h)$ where h is the maximum mesh size of the triangulation. A disadvantage of this method is the degeneracy of the implicit equation that arises because the projection matrix has a zero eigenvalue. The elliptic regularity is lost and issues on the numerical stability arise. Moreover, complicated terms have to be integrated over triangular domains.

Even though embedding methods are used more often they have limitations. The known convergence analysis is restricted to submanifolds of codimension one. When the extension is restricted to a band suitable boundary conditions have to be formulated. Moreover, the bandwidth must be less than the local radii of curvature which leads to complications at corners or edges.

The present thesis introduces three new embedding methods. The first one deals with function approximation on arbitrary domains. The so called *Ambient B-spline Method* creates a C^k-approximation for arbitrary $k \in \mathbb{N}$ to a given function or discrete function values on a smooth compact submanifold. There is no restriction on the dimension, the codimension or the genus of the submanifold. As the name indicates the method returns a spline. The workflow of the method is simple and straightforward: The given function is extended to some ambient domain of the submanifold where we use tensor product B-splines on the Cartesian grid to approximate the function. Then the result is restricted to the submanifold again. The advantage of using Cartesian grid is that we can use well-studied numerical techniques with small error and accurate stability measures. We can show that the Ambient B-spline Method shows optimal error convergence for manifolds of codimension one. The Ambient B-spline Method can even be used to model surfaces when a suitable reference manifold is given as domain (see [Leh13]).

The second and third method handle intrinsic linear elliptic PDEs on submanifolds. The *Ambient Signed Distance Method* works similar to the earlier mentioned method by Dzuik and Elliott. We overcome the drawback that ellipticity is lost by avoiding any projection. The given intrinsic PDE is translated into a PDE in an ambient domain. The formulation of the embedding PDE makes use of the signed distance function of the submanifold. The structure of the ambient domain,

the translation of the PDE and a new version of boundary conditions are key characteristics of the Ambient Signed Distance Method. The so constructed PDE on the ambient domain is then solved with the Finite Element Method. Here, we use the tensor product spline space as finite element space. The simple structure of tensor product B-splines as well as their stability and approximation properties are preserved. The method results a spline of arbitrary order $n \in \mathbb{N}$ defined on the ambient domain that solves the embedding PDE and is constant in normal direction of the submanifold. Presented numerical results show optimal error behaviour with a convergence in $\mathcal{O}(h^n)$ where h is the grid width of the underlying grid. The implementation is simple. Moreover, we avoid any scattered data problem and all integration domains are multi-dimensional intervals.

The basic concept of the *Ambient Level Set Method* is the same as in the Ambient Signed Distance Method. The fundamental difference is that the signed distance function is not needed when constructing the embedding PDE. Instead, any level set function of the submanifold suffices. The intrinsic PDE is extended to an embedding PDE of the form

$$\sum_{i,j} A_{ij} \cdot \frac{\partial U}{\partial x_i \partial x_j} + \sum_i B_i \cdot \frac{\partial U}{\partial x_i} + CU = F,$$

for a constant $C \in \mathbb{R}$ and a function F. The matrix $A(x,t)$ and the vector $B(x,t)$ depend on the level set and are computed via ordinary differential equations. This way of formulating the embedding PDE can easily be applied to any linear elliptic second-order PDE on a compact submanifold. It can also be generalised to parabolic PDEs. Just like the Ambient Signed Distance Method here the equations are solved using the Finite Element Method. Again, due to the structure of the ambient domain the integration domains are multidimensional intervals. This method, as well, returns a spline. Numerical tests show that the result converges to the exact solution as the grid width converges to zero.

Both methods are constructed in such a way that they solve static linear elliptic second-order PDEs. Both concepts can be transferred to parabolic PDEs as well. For that, we can use any time-stepping method. There is no need of reextensions. Therefore, many PDEs that appear in applications are covered by these methods.

Throughout the whole thesis we concentrate on the static elliptic second-order PDE

$$-\triangle_\omega u + cu = f$$

for a constant $c > 0$ and a continuous function f defined on the submanifold ω. By $\triangle_\omega$ we denote the Laplace-Beltrami operator.

We can derive simple, accurate, robust and elegant implementations of all three methods. The implementations are done in MATLAB and numerical examples are illustrated in this thesis. All three implementations take as input an implicitly represented submanifold. If the original problem is not already in implicit form but triangulated, we can use a number of algorithms that achieve this representation given a triangulated input. See for example [YT99].

The thesis is organised in the following way:

Chapter 2 shortly reviews some common tools that are needed throughout the thesis. We introduce standard Sobolev spaces. The general procedure of the Finite Element Method is also described. We give the definition of the intrinsic gradient as well as the Laplace-Beltrami operator. Finally, we introduce the model PDE of this thesis.

Chapter 3 deals with the most important tool of the methods, namely splines. In this chapter we define splines and B-splines. Also tensor product B-splines are introduced and analysed. Some analysis on their stability and approximation order is done.

We present the Ambient B-spline Method in Chapter 4. The focus is given to the theoretical proof of its error convergence. Here, we establish that the error of the resulting spline converges in $\mathcal{O}(h^n)$ where h is the grid width and n is the order of the spline. We give a short description of the implementation. Finally, some 3-D examples are presented. These numerical examples confirm the theory.

Chapter 5 deals with the Ambient Signed Distance Method. The detailed workflow of the method is spread over three sections. After an explanation of the implementation we show numerical results.

The Ambient Level Set Method is introduced in Chapter 6. Here the focus lies on the underlying idea, that is, on the derivation of the embedding PDE. An implementation and numerical results are also given in this chapter.

The last chapter is a conclusion that summerises the main results and discusses some open issues.

2 Preliminaries

In this thesis we present three methods. One deals with function approximation on manifolds and the other two solve PDEs on manifolds. Before we introduce these methods in the following chapters we recapitulate some tools here: standard Sobolev spaces, the Finite Element Method and intrinsic differential operators. Moreover, we introduce the model PDE which is used throughout the thesis to illustrate certain aspects.

2.1 Sobolev Spaces

We will introduce Sobolev spaces as they are commonly defined. The Sobolev space $W_p^n(\Omega)$ with $n \in \mathbb{N}$ and $1 \leq p \leq \infty$ is a vector space of functions on the open set Ω. This vector space is equipped with a norm that is a combination of the standard L_p-norm of the function itself and its derivatives up to order n. We will see in Chapter 4 how to define Sobolev spaces on manifolds. Here, we limit ourselves to standard Sobolev spaces.

Let Ω be an open subset of $\mathbb{R}^d$ with piecewise smooth boundary. Let f be a Lebesgue integrable function on Ω and define for $1 \leq p < \infty$ the L_p-norm as

$$\|f\|_{L_p(\Omega)} := \left(\int_\Omega |f(x)|^p \, dx \right)^{\frac{1}{p}} .$$

For $p = \infty$ we have

$$\|f\|_{L_\infty(\Omega)} := \operatorname*{ess\,sup}_{x \in \Omega} |f(x)| .$$

The functional space $L_p(\Omega)$ is then defined as

$$L_p(\Omega) = \{ f : \Omega \to \mathbb{R} \mid \|f\|_{L_p(\Omega)} < \infty \} .$$

For functions in L_p the weak derivative is defined as follows:

Definition 1. *Let $f \in L_p(\Omega)$ and $\alpha = (\alpha_1, \ldots, \alpha_d)$ be a multi-index. Then $g = \partial^\alpha f$ is called the* weak derivative of f of order α *if*

$$\int_\Omega f(x)\, \partial^\alpha \phi(x)\, dx = (-1)^{|\alpha|} \int_\Omega g(x)\, \phi(x)\, dx$$

for any test function $\phi \in C_0^\infty(\Omega)$.

With the help of the weak derivative we can now define the following norm

$$\|f\|_{W_p^n(\Omega)} := \begin{cases} \left(\sum_{|\alpha| \leq n} \|\partial^\alpha f\|_{L_p(\Omega)}^p \right)^{\frac{1}{p}} & \text{if } p < \infty \\ \max_{|\alpha| \leq n} \|\partial^\alpha f\|_{L_\infty(\Omega)} & \text{if } p = \infty \end{cases}$$

The Sobolev space $W_p^n(\Omega)$ is defined as

$$W_p^n(\Omega) := \{f \in L_p(\Omega) \mid \partial^\alpha f \in L_p(\Omega),\ |\alpha| \leq n\}\,.$$

With the norm $\|\cdot\|_{W_p^n(\Omega)}$ the Sobolev space is a Banach space. The case $p = 2$ is special. Together with the inner product

$$\langle f, g\rangle = \int_\Omega f(x)g(x) \quad \text{for } f,\ g \in L_2(\Omega)$$

$L_2(\Omega)$ becomes a Hilbert space with the norm

$$\|f\|_{L_2(\Omega)} = \sqrt{\langle f, f\rangle}.$$

Therefore, $L_2(\Omega)$ is also denoted by $H(\Omega)$. Similarly, the Sobolev space $W_2^n(\Omega)$ is also denoted by $H^n(\Omega)$.

2.2 Finite Element Method

The Finite Element Method (FEM) is a technique for finding approximate solutions in the calculus of variations and for elliptic differential equations. The basic idea of Finite Element Methods is to solve a given problem in a suitable finite dimensional subspace. Usually, this subspace is indicated with the subscript h. Here, h is a discretising parameter. The solution of the FEM converges to the exact solution as h tends to 0.

Let V be a linear space. Given the following minimisation problem:

$$\min \frac{1}{2}\, a(v, v) - \lambda(v), \quad v \in V$$

where $a\colon V \times V \to \mathbb{R}$ is a symmetric, positive bilinear form and $\lambda\colon V \to \mathbb{R}$ is a linear functional. We find the minimum in $u \in V$ if and only if u solves

$$a(u, v) = \lambda(v) \quad \forall v \in V.$$

See [Bra97] for a proof and details about existence and uniqueness conditions of the solution.

Now, let S_h be an N-dimensional subspace of V and $\{\zeta_1, \ldots, \zeta_N\}$ a basis of S_h. Then, u_h is called a solution in S_h if

$$a(u_h, v_h) = \lambda(v_h), \quad \forall v_h \in S_h.$$

This is equivalent to u_h being the solution of

$$a(u_h, \zeta_i) = \lambda(\zeta_i), \quad i = 1, \ldots, N.$$

With

$$u_h = \sum_{i=1}^{N} p_i \zeta_i$$

for some coefficients $p_i \in \mathbb{R}$ we obtain the system of linear equations

$$Ap = \Lambda$$

with $A_{ij} = a(\zeta_i, \zeta_j)$ and $\Lambda_i = \lambda(\zeta_i)$ for $i, j = 1, \ldots, N$. We solve this linear system of equations to get u_h.

The Céa-Lemma shows that the error of the solution obtained by the FEM can be estimated by the error of the best approximation of u in the subspace S_h.

Lemma 1. *(Céa-Lemma) Let V be a Hilbert space and $S_h \subset V$. Let $a\colon V \times V \to \mathbb{R}$ be bilinear form with the following properties:*

- $a(u,u) \geq \alpha \|u\|_{H^m}^2$ *for all $u \in V$ and a constant $\alpha > 0$ (continuity)*
- $|a(u,v)| \leq C\|u\|_{H^m}\|v\|_{H^m}$ *for all $u, v \in V$ and a constant $C > 0$ (coercivity).*

Let u and u_h be as before. Then

$$\|u - u_h\|_{H^m} \leq \frac{C}{\alpha} \inf_{v_h \in S_h} \|u - v_h\|_{H^m} .$$

A proof can be found in [Bra97]. This lemma says that the solution of the FEM is as good as the best approximation in S_h, up to a constant. Therefore, by choosing a suitable subspace S_h we can obtain arbitrarily good approximations by the FEM.

2.3 Intrinsic Differential Operators

Now, we will give the definition of the intrinsic gradient and the Laplace-Beltrami operator as given in [DE13]. Let $\omega \subset \mathbb{R}^d$, $d \in \mathbb{N}$, be an $(d-1)$-dimensional submanifold such that for every point $x_0 \in \omega$ there exists a local parametrisation. That means for every point $x_0 \in \omega$, we find an open set $U \subset \mathbb{R}^d$ containing x_0, an open connected set $V \subset \mathbb{R}^{d-1}$ and a k-times differentiable map $\Phi\colon V \to U \cap \omega$ for $k \in \mathbb{N} \cup \{\infty\}$ such that Φ is bijective and $\operatorname{rank} \nabla\Phi = d-1$. A C^k-atlas is a collection of local parametrisations $\Phi_i\colon V_i \to U_i \cap \omega, i \in I$ such that $\bigcup_i \Phi_i(V_i) = \omega$.

Let $G\colon V \to \mathbb{R}$ be the first fundamental form at $X_0 = \Phi^{-1}(x_0)$:

$$G_{ij}(X_0) := \frac{\partial\Phi(X_0)}{\partial X_i} \cdot \frac{\partial\Phi(X_0)}{\partial X_j} .$$

Let $f : \omega \to \mathbb{R}$ be a twice differentiable function, i.e., for all local parametrisations Φ_i the composition $f \circ \Phi_i$ is twice differentiable. Then the *Laplace-Beltrami operator* of f on ω is given by

$$\triangle_\omega f(x_0) := \frac{1}{\sqrt{|G(X_0)|}} \sum_{i,j=1}^{d-1} \frac{\partial}{\partial X_j} \left(G_{ij}^{-1}(X_0)\sqrt{|G(X_0)|}\, \frac{\partial(f\circ\Phi)}{\partial X_i}(X_0) \right)$$

where $|G(X_0)| = \det G(X_0)$. The *intrinsic gradient* of f on ω is defined as

$$\nabla_\omega f(x_0) := \sum_{i,j=1}^{d-1} G_{ij}^{-1}(X_0) \frac{\partial(f\circ\Phi)}{\partial X_j}(X_0) \frac{\partial\Phi}{\partial X_i}(X_0) .$$

The intrinsic gradient as well as the Laplace-Beltrami operator are independent of the parametrisation. Now, let $\omega \subset \mathbb{R}^d$ be a C^1-hypersurface. That means, for every point $x_0 \in \omega$ there exists an open set $U \subset \mathbb{R}^d$ with $x_0 \in U$ and a differentiable function $\varphi\colon \mathbb{R}^d \to \mathbb{R}$ with $\nabla\varphi \neq 0$ and

$$U \cap \omega = \{x \in U \mid \varphi(x) = 0\} .$$

For a differentiable function $f : \omega \to \mathbb{R}$ we define the projected gradient as

$$\nabla_P f(x) := \nabla\bar{f}(x) - \nabla\bar{f}(x) \cdot \nu(x)\nu(x)$$

where $\bar{f}$ is any smooth extension of f to a d-dimensional neighbourhood of ω and $\nu = \frac{\nabla\varphi}{|\nabla\varphi|}$. We note that the projected gradient is not restricted to points x on the manifold.

Let $D_i f(x)$ denote the i-th entry of $\nabla_P f(x)$. Then

$$\triangle_P f := \nabla_P \cdot \nabla_P f = \sum_{i=1}^{d} D_i D_i f \ .$$

In their paper [DE13] Dziuk and Elliott establish the equivalence of intrinsic and projected operators, i.e.,

$$\nabla_\omega f(x) = \nabla_P f(x) \quad \text{and} \quad \triangle_\omega f(x) = \triangle_P f(x), \quad \text{for } x \in \omega.$$

2.4 Second-Order Elliptic Partial Differential Equations

We give a short insight into the huge field of Partial Differential Equations (PDEs). We introduce different types and set the model PDE of this thesis.

PDEs can be divided into different types. In general, a linear second-order PDE in d variables is of the form

$$\sum_{i,j=1}^{d} a_{ij}(x) \frac{\partial^2 u}{\partial x_i \partial x_j} + \sum_{i=1}^{d} b_i(x) \frac{\partial u}{\partial x_i} + c(x)\, u = f(x)$$

where $x = (x_1, \ldots, x_d)$ lies in some domain $\Omega \subseteq \mathbb{R}^d$. We assume $u \in C^2(\Omega)$ and we have the symmetry relation

$$a_{ij}(x) = a_{ji}(x)\,.$$

The linear differential operator

$$L(x, u, \nabla u, \nabla^2 u) = \sum_{i,j=1}^{d} a_{ij}(x) \frac{\partial^2 u}{\partial x_i \partial x_j} + \sum_{i=1}^{d} b_i(x) \frac{\partial u}{\partial x_i} + c(x)\, u$$

is called elliptic at the point x_0 if the $d \times d$-matrix

$$A(x) = (a_{ij}(x))$$

is positive definite at x_0. A PDE $L(x, u, \nabla u, \nabla^2 u) = f$ is called *elliptic* if the matrix $A(x)$ is elliptic at all points $x \in \Omega$. If $A(x)$ has one negative and $d-1$ positive eigenvalues we call the PDE *hyberbolic*. A PDE is called *parabolic* if $A(x)$ is positive semi-definite but not positive definite and $\operatorname{rank}(A(x), b(x)) = d$.

The following theorem on the uniqueness of solutions can be found in [GT01]:

Theorem 1. *Let Ω be an open subset of $\mathbb{R}^d$. Let the linear differential operator L be elliptic in Ω with $c \leq 0$. Suppose u and v are functions in $C^2(\Omega) \cap C^0(\overline{\Omega})$ with $L(x, u, \nabla u, \nabla^2 u) = L(x, v, \nabla v, \nabla^2 v)$ in Ω and $u = v$ on $\partial\Omega$. Then $u = v$ on Ω.*

The theorem results from the maximum principle.

Throughout this thesis we limit to second-order elliptic PDEs. Specifically, we illustrate some aspects by means of the model PDE

$$-\triangle u + cu = f$$

where $\triangle$ is the Laplace operator, $c > 0$ is a constant and $f \in H(\Omega)$. Obviously, the model PDE is elliptic because

$$-\triangle u + cu = f \quad \Leftrightarrow \quad \triangle u - cu = -f \ ,$$

and then $A(x) = \mathbb{1}$.

3 Splines

Polynomials were regarded as a good tool for approximation. They are easily evaluated and differentiated and the polynomial space $\mathbb{P}^n$ of order $n \in \mathbb{N}$ is a finite dimensional linear space. Moreover, the Weierstraß theorem says that any continuous function on some interval $[a, b]$ can be approximated arbitrarily good by polynomials. Yet, for this one might need a very high order n. Apart from that, the standard basis functions of the polynomials, the monomials, have global support. A small variation in the input data results in a recalculation of the whole system. Therefore, approximation with polynomials is more of theoretical interest than practical use.

In 1946 Schoenberg introduced *splines*. These are piecewise polynomials based on a partition of some interval $[a, b]$. Even though piecewise polynomials are not as smooth as polynomials, they provide many advantages and therefore turn out as a very useful tool in approximation theory. The spline space of order n is a finite dimensional linear space. In case of a simple partition splines of order n are in C^{n-2}. The compact support of the standard basis, the so called *B-splines*, is an additional benefit. Moreover, splines are stable. That means, we find a nice relation between the coefficients of a spline and the spline itself. Splines provide optimal approximation order (see Section 3.3). For practical use it is also good that splines are easy to store and evaluate on a computer. Finally, the derivatives of splines are splines as well.

In this chapter we will define B-splines (see Section 3.1) as well as tensor product B-splines (Section 3.2) and analyse some of their properties. In Section 3.3 we summarise some existing results of splines on different domains. For a more comprehensive insight into splines and B-splines we refer to [Sch80], [Möß06] and [HH13].

3.1 B-Splines

Splines are piecewise polynomial functions based on a finite, infinite or bi-infinite knot sequence. Now, we will introduce splines together with their standard basis, the B-splines. B-splines can be defined in different ways. Here, we define them by Marsden's identity. In this section we also name some useful properties of splines and B-splines.

Let $T = \{\tau_i\}_{i \in I}$ with knots $\tau_i \in \mathbb{R}$ be a monotonically increasing knot sequence of order $n \in \mathbb{N}$. The index set I can be finite, i.e., $I = 1, \dots, m + n$. Then we call m the dimension. The index set I can also be infinite or bi-infinite, i.e., $I = \mathbb{N}$ or $I = \mathbb{Z}$, respectively. The knot sequence is called *non-degenerate* if every knot has a multiplicity $\leq n$. In case of a finite knot sequence we additionally claim

$$\tau_n < \tau_{n+1} \quad \text{and} \quad \tau_m < \tau_{m+1}\,.$$

In case of a finite knot sequence the domain is usually set as $D(T) := [\tau_n, \tau_{m+1})$. Otherwise, we have $D(T) := (\inf T, \sup T)$.

Let τ_i and $\tau_{i+1} \in T$ with $\tau_i < \tau_{i+1}$, then each spline $s : D(T) \to \mathbb{R}$ of order n equals a polynomial of order n on the interval $[\tau_i, \tau_{i+1})$. At simple knots the spline is $(n-2)-$times differentiable and at knots of multiplicity $k \in \mathbb{N}$ the spline s is $(n-k-1)-$times differentiable. The set of all splines s of order n based on this knot sequence is called the *spline space* $S^{n,T}(D(T))$.

In the following, let T be a diverging bi-infinite knot sequence where all knots are of multiplicity one. Then, $D(T) = \mathbb{R}$ and $S^{n,T}(\mathbb{R}) \subset C^{n-2}(\mathbb{R})$. From now on, we denote the spline space $S^n := S^{n,T}(\mathbb{R})$ for the sake of readability.

We will now define B-splines which are the standard basis of the spline space. B-splines can be defined in several ways, either by using a recursive formula or with the help of divided differences. Here, we define B-splines by introducing Marsden's identity.

Definition 2. *Let $T = \{\tau_i\}_{i\in\mathbb{Z}}$ be a bi-infinite non-degenerate knot sequence and $n \in \mathbb{N}$. Let $\psi_j^n \colon \mathbb{R} \to \mathbb{R}$ be defined as*

$$\psi_j^n(x) := \prod_{k=1}^{n-1} (\tau_{j+k} - x) \quad \text{for } j \in \mathbb{Z}.$$

The functions $b_j^n,\ j \in \mathbb{Z}$ defined by

$$\operatorname{supp} b_j^n \subseteq [\tau_j, \tau_{j+n}]$$

and

$$(t-x)^{n-1} = \sum_j b_j^n(t)\, \psi_j^n(x) \quad \text{for all } t,\ x \in \mathbb{R}$$

are called B-splines of order *n*.

Marsden's identity gives the existence of such functions b_j^n. Their uniqueness, up to scaling, is determined by their support.

Theorem 2 (Marsden's identity)**.** *Let T be a non-degenerate knot sequence of a spline space of order $n \in \mathbb{N}$ and let $\psi_j^n,\ j \in \mathbb{Z}$ be defined as in Definition 2. Then there exist functions $b_j^n \colon \mathbb{R} \to \mathbb{R}$ such that*

$$(t-x)^{n-1} = \sum_j b_j^n(t)\, \psi_j^n(x) \quad \text{for all } t,\ x \in \mathbb{R}.$$

The B-splines of order n form a basis of the spline space S^n. That means, B-splines are linearly independent,

$$\sum_j \xi_j\, b_j^n(t) = 0 \quad \Longleftrightarrow \quad \xi_j = 0 \quad \text{for } t \in \mathbb{R},\ j \in \mathbb{Z}$$

and each spline $s \in S^n$ can be represented as

$$s(t) = \sum_j \xi_j b_j^n(t) =: \xi\, B.$$

A proof can be found in [Möß06].

The coefficients $\xi_j \in \mathbb{R}$ are called *control points*. The control points can also be multi-dimensional, i.e., $\xi_j \in \mathbb{R}^d$ for some $d \in \mathbb{N}$. Then, we obtain a spline curve $s \colon \mathbb{R} \to \mathbb{R}^d$. Throughout this thesis we set $\xi_j \in \mathbb{R}$ if not specified otherwise. The linear connection of the control points is called *control polygon*.

The B-spline basis has many advantages, compared with other bases like for example the truncated power basis. B-splines are non-negative and form a partition of unity,

$$\sum_j b_j^n(t) = 1\,.$$

Moreover, splines are variation diminishing. While a polynomial approximation tends to oscillate around the original values the deflection of a spline is always less than or equal the one of the corresponding control polygon. Apart from that, a spline curve lies within the convex hull of its control polygon and is affine invariant.

A B-spline of order n can be derived as a combination of B-splines of order $n-1$. This results in a recursive formula.

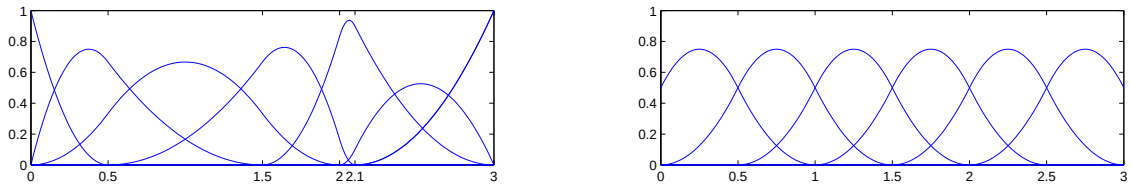

Figure 3.1: B-splines of order three. *left:* arbitrary knot sequence, *right:* uniform knot sequence.

Theorem 3. *Let*

$$w_j^n(t) := \frac{t - \tau_j}{\tau_{j+n} - \tau_j}\,.$$

Then

$$\begin{aligned} b_j^1 &= \chi([\tau_j, \tau_{j+1})) \\ b_j^{n+1} &= w_j^n b_j^n + \left(1 - w_{j+1}^n\right) b_{j+1}^n\,, \end{aligned}$$

where $\chi : \mathbb{R} \to \{0, 1\}$ *is the characteristic function.*

The de Boor algorithm uses this recursion formula to evaluate splines. It operates on the control points. It is easy to implement, fast and numerically stable. See [Sch80] for a description of the de Boor algorithm. Splines can be easily stored and evaluated on a computer. This makes splines attractive for applications.

The derivative of a B-spline of order n is a linear combination of B-splines of order $n - 1$. Therefore, differentiation of splines in B-spline-form can also be derived by operations on the control points.

The main advantage of B-splines is their compact support: The B-spline b_j^n depends only on the knots $\tau_j, \ldots, \tau_{j+n}$. Therefore, a change of some data of an approximation problem can be handled by a recalculation of the corresponding control points. Moreover, the locality of B-splines enables hierarchical splines, see [Leh13].

If the knot sequence is uniformly spaced, i.e.,

$$\tau_{j+1} - \tau_j = h \qquad \text{for all } j \text{ and a fixed } h \in \mathbb{R}$$

then the B-splines are called *uniform B-splines*. Uniform B-splines are translations of each other (see Figure 3.1 right).

Apart from all these positive properties B-splines are *stable*. That means small perturbations in the control points imply only small changes of the spline and vice versa. In Section 3.3 we analyse stability as well as the approximation behaviour of splines. Before that we introduce tensor product splines which are multidimensional splines.

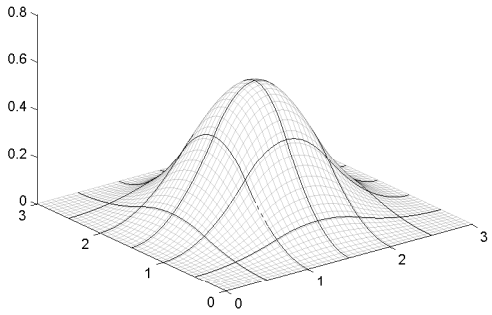

Figure 3.2: a two-dimensional tensor product B-spline of order $n = [3,3]$

3.2 Tensor Product Splines

The Ambient B-spline Method translates a given approximation problem into a higher dimensional space. Therefore, we have a look at tensor product splines which are generalisations of splines to approximate functions of several variables. Tensor product splines are based on a rectangular mesh. Another option are triangular splines. These are multidimensional splines that are defined over arbitrary planar triangulations. The construction of a rectangular mesh is less complicated than constructing a triangulation. Since tensor product B-splines fulfil all requirements needed for the Ambient B-spline Method we operate on them.

Let $d \in \mathbb{N}$ and $T_1, \dots, T_d$ be non-degenerate knot sequences. As above, we let them be bi-infinite and diverging for simplicity. The d-dimensional spline space of order $n = [n_1, \dots, n_d]$ is defined as

$$S^{n,T}(\mathbb{R}^d) := \bigotimes_{\ell=1}^{d} S^{n_\ell, T_\ell}(\mathbb{R}),$$

where the grid T is the Cartesian product of the univariate knot sequences $T_\ell = \{\tau_i^\ell\}_{i\in\mathbb{Z}}$ with $\ell = 1, \dots, d$. A *tensor product B-spline* $b_k^n \in S^{n,T}(\mathbb{R}^d)$ is defined as

$$b_k^n(t) := \prod_{j=1}^{d} b_{k_j}^{n_j}(t_j), \qquad k \in \mathbb{Z}^d,\ t \in \mathbb{R}^d.$$

Its support equals

$$\operatorname{supp}(b_k^n) = \left[\tau_{k_1}^1, \tau_{k_1+n_1}^1\right] \times \cdots \times \left[\tau_{k_d}^d, \tau_{k_d+n_d}^d\right].$$

A tensor product spline is then a linear combination of tensor product B-splines:

$$s(t) = \sum_{k\in\mathbb{Z}^d} \xi_k b_k^n(t), \quad t \in \mathbb{R}^d$$

with control points $\xi_k \in \mathbb{R}$. As in the one-dimensional case the control points can be multi-dimensional as well.

General properties of the tensor product B-splines can be derived from the properties of the one-dimensional B-splines. We refer to [Sch80] for more details. One of the most favourable

properties of tensor product B-splines is the stability in $\mathbb{R}^d$. A proof using the multivariate de Boor-Fix functionals can be found in [Möß06].

Yet, on arbitrary domains $\Omega \subset \mathbb{R}^d$ stability is typically lost. The reason is that the support of a tensor product B-spline might have arbitrarily small intersections with Ω. Also the constant in the estimate of the approximation error depends on the aspect ratio of the grid cells. In the next section we show how these problems can be overcome.

3.3 Stability and Approximation Power

In this section we summarise some results on stability and approximation power of splines. While splines give optimal approximation behaviour on $\mathbb{R}^d$ they have limitations on arbitrary domains $\Omega \subset \mathbb{R}^d$. Modifications of the standard tensor product B-splines solve most of the arising issues. Yet, on manifolds tensor product spline approximation does not work in the direct way.

On $\mathbb{R}$ B-splines have many favourable properties. One of them is *stability*: The spline is bounded by the size of the coefficients and at the same time the control points are closely related to the values of the spline. So, small changes in the control points result in small changes of the spline and vice versa. Formally, we say a basis is *stable* or *well conditioned* if there exist constants $c_1, c_2 > 0$ such that

$$c_1 \, \|\xi\|_\infty \leq \|s\|_\infty \leq c_2 \, \|\xi\|_\infty$$

where ξ are the control points of the spline s. The constants do not depend on the grid width but only on the order $n \in \mathbb{N}$. The ratio $\frac{c_2}{c_1}$ is called the *conditioning* of the basis. A proof of the stability of B-splines on $\mathbb{R}$ is given in [Möß06].

Moreover, B-splines have optimal approximation power on $\mathbb{R}$. That means for any $f \in W_p^n(\mathbb{R})$ there exists a spline $s \in S^n(\mathbb{R})$ such that

$$\|s - f\|_{W_p^m(\mathbb{R})} \leq c \, h^{n-m} \|f\|_{W_p^n(\Omega)} \quad \text{for } m < n$$

where h is the maximal grid width and c is a constant depending on n and p. We refer to [Sch80] for a proof. The stability and approximation properties of splines on $\mathbb{R}$ can be transferred to tensor product splines on $\mathbb{R}^d$ without further complications.

On intervals or multidimensional intervals stability is given as well. Here, the knots can be set such that the domain equals a union of grid cells. With this choice of knots even the error analysis reduces to the case of polynomial approximation on boxes. The Bramble-Hilbert lemma gives an estimate on polynomial approximation. The here presented simplified form can be derived directly from the classical Bramble-Hilbert Lemma:

Lemma 2. *Let Ω be a d-dimensional interval of size h^d. Let α and β be multi-indices with $\beta_i < \alpha_i$ for $i = 1, \ldots, d$, $|\alpha| = n$ and $|\beta| = m$. For any function $f \in W_p^n(\Omega)$ there exists a polynomial $\Pi \in \mathbb{P}^n$ such that*

$$\|f - \Pi\|_{W_p^m(\Omega)} \leq c \left(\sum_{i=1}^{d} h^{\alpha_i - \beta_i} \|\partial_i^{\alpha_i} f\|_{L_p(\mathbb{R})} \right),$$

with c depending on n and d.

From this it follows that tensor product B-splines show optimal approximation behaviour on domains that consist of grid cells.

Corollary 1. *Let* $\Omega = \bigcup_{k\in K} \left[\tau^1_{k_1}, \tau^1_{k_1+1}\right] \times \ldots \times \left[\tau^d_{k_d}, \tau^d_{k_d+1}\right]$ *for an index set* $K \subseteq \mathbb{Z}^d$ *be the union of some grid cells. Let* $f \in W^n_p(\Omega)$*. Then there exists a spline* $s \in S^n(\Omega)$ *such that*

$$\|f - s\|_{W^m_p(\Omega)} \leq c\, h^{n-m} \|f\|_{W^n_p(\Omega)}, \quad \text{for } m < n,$$

where the constant c *depends on* n *and* d*.*

On arbitrary domains $\Omega \subset \mathbb{R}^d$ both these properties, stability and full approximation power, can not be guaranteed. In 1980 Dahmen, De Vore and Scherer presented some results on this topic in their paper [DDS80]. They prove that on Lipschitz graph domains Ω we have for $f \in W^n_p(\Omega)$

$$\inf_{\Pi\in\mathbb{P}^{n-1}} \|f - \Pi\|_{W^m_p(\Omega)} \leq c \sum_{|\alpha|=n} \|\partial^\alpha f\|_{L_p(\Omega)}, \quad m < n$$

where c is a constant depending on n, p and Ω. The same result holds on generalised graph domains as Reif showed in [Rei12]. The definition of generalised graph domains as well as the explicit value of the constant c are given in that paper.

There exist several attempts to handle stability of B-splines on arbitrary domains. Weighted extended B-splines (web-splines) are introduced in [HRW01] and normalised B-splines are suggested by Mößner in [Möß06]. In [MR08b] Mößner and Reif specified that the stability loss is caused by B-splines which do not fulfil the property of being α-proper.

Definition 3. *Let* $\Omega \subset \mathbb{R}^d$*. A tensor product B-spline* $b^n_i, i \in \mathbb{Z}^d, n \in \mathbb{N}$ *is called* α-proper *if there exist boxes* $P_i = \left[p^{1,1}_i, p^{1,2}_i\right] \times \cdots \times \left[p^{d,1}_i, p^{d,2}_i\right]$ *and* $R_i = \left[r^{1,1}_i, r^{1,2}_i\right] \times \cdots \times \left[r^{d,1}_i, r^{d,2}_i\right]$ *with*

- $R_i \subset \operatorname{supp} b^n_i \cap \Omega \subset P_i$
- *the vector of edge lengths are related by* $\alpha \left(r^{j,2}_i - r^{j,1}_i\right) = p^{j,2}_i - p^{j,1}_i$ *for all* $j = 1, \ldots, d$
- *the box* P_i *has one corner in common with* $\operatorname{supp} b^n_i$*.*

See [MR08b] for more detailed notes and a simple example on this topic. Omitting B-splines that are not α-proper is another option to guarantee stability. Yet, this influences the approximation power.

Another issue that arises in case of tensor product B-splines on an arbitrary domain Ω is that the constant in the error estimate depends on the aspect ratio of the grid cells in T^d. Different grid widths in different directions can influence the approximation in a negative way. Condensed B-splines were introduced in [Sis11] to handle this topic in the bivariate case. Here, knots outside the domain are replaced such that the support of the B-spline is minimised. In that way stability is preserved and full approximation power is obtained.

Approximation on manifolds with tensor product B-splines is even more complicated. If the given data lies on a submanifold in $\mathbb{R}^d$ and one uses d-dimensional tensor product B-splines to approximate the data then the result is usually not satisfying. Let p_i be the data points and $F_i = f(p_i)$ the data values. The matrix B with $B_{ij} = b^n_j(p_i)$ has very bad condition number which influences the accuracy of the approximation. It might even happen that the linear system of equations $B\xi = F$ is singular. Geometrically spoken, there might exist a number of splines approximating the given values at the points in the submanifold. Therefore, a direct approximation is not possible. Different attempts to approximate data on manifolds are mentioned in Chapter 1.

We will see in Section 4.1 that the Ambient B-spline Method smartly prevents all of these issues. We extend the data given on the submanifold to some ambient domain. We construct a uniform

grid with the same grid width in all directions and therefore avoid a dependence of the constant in the error estimate on the grid. Moreover, we do not choose Ω arbitrarily. The ambient domain Ω is a union of grid cells and each cell is uniformly filled with data points. So, we can guarantee that at least $\left(\frac{1}{n}\right)^d$ of the whole support of each B-spline is filled with data points. Here, $n = n_1 = \cdots = n_d$ is the order of the tensor product B-splines. So, the earlier mentioned stability problem is avoided.

4 Ambient B-spline Method

The Ambient B-spline Method provides a novel technique to approximate arbitrary functions given on submanifolds. By the Ambient B-spline Method C^k approximations for arbitrary $k \in \mathbb{N}$ are constructed. That means, the approximation can be chosen to be as smooth as the user wishes supposed the submanifold is sufficiently smooth.

The Ambient B-spline Method is an embedding method. It works for any compact submanifold that is embedded in some higher dimensional space. The basic idea of the method is illustrated in Figure 4.1. Let ω denote the submanifold embedded in $\mathbb{R}^d, d \in \mathbb{N}$, and $f : \omega \to \mathbb{R}^m$ be a given function for some $m \in \mathbb{N}$. First, the function f is extended onto some domain $\Omega \subset \mathbb{R}^d$ in the ambient space of the submanifold. Then, the extended function F is approximated in Ω. Finally, the result is restricted to ω. In this way the given problem is translated into an approximation problem in $\mathbb{R}^d$. This is a well known and studied topic for which there exist already many different solution methods.

Obviously, this workflow leaves still space for variation: The way to extent the function f and the methods of approximation in Ω can still be chosen. A general way of extending f is to copy the function values in some direction that is not tangential to the submanifold ω. We choose the extension F to be constant in normal direction of the submanifold. This provides some useful properties of the extension and simplifies the implementation.

The way we approximate the extended function gives the method its name: We use an approximation by tensor product B-splines based on a uniform grid. B-splines are easy to handle and have optimal approximation order (see Chapter 3).

The Ambient B-spline Method has certain advantages. It is easy to implement and does not require any parametrisation of the submanifold. We do not have to construct complicated manifold spaces nor do we have to triangulate the submanifold. Furthermore, the presented method is not restricted to submanifolds of a certain dimension or genus, but works for any smooth and compact submanifold without selfintersections. The Ambient B-spline Method transforms the problem to a situation in $\mathbb{R}^d$ where approximation is already well studied. We prove that the method has optimal error behaviour. Yet, we will see in Section 4.5 that the proof we present cannot be translated to submanifolds of codimension greater than one. For now, we concentrate on submanifolds of codimension one.

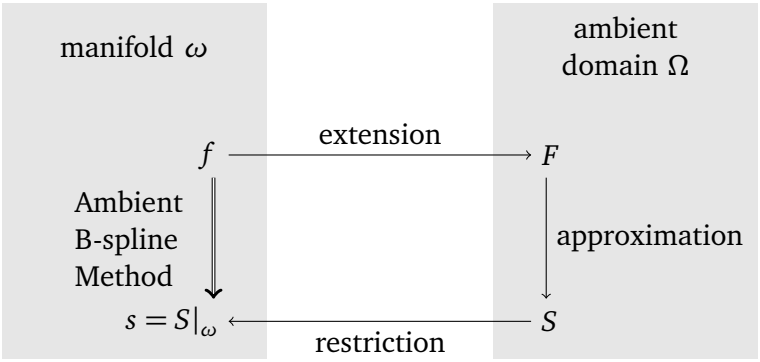

Figure 4.1: workflow of the Ambient B-spline Method

In Section 4.1 we explain the concept of the Ambient B-spline Method in detail. Section 4.2 provides a theoretical analysis on the approximation order. A possible implementation is described in Section 4.3 and Section 4.4 shows some numerical results. Finally, the reason why we limit to submanifolds of codimension one is given in Section 4.5.

4.1 Concept of the Method

We have now gained an idea of the workflow of the Ambient B-spline Method and some knowledge about its main tool, namely B-splines (see Chapter 3). In this section we analyse each step of the algorithm in detail.

Let us assume that a smooth compact embedded submanifold $\omega \subset \mathbb{R}^d$ of codimesion one and a function $f : \omega \to \mathbb{R}^m$ for $m \in \mathbb{N}$ are given. For now, we concentrate on the case $m = 1$.

First of all, we need to define the ambient domain Ω to which we want to extend the function f. Therefore, we choose a grid width h and construct a regular mesh G that covers the whole submanifold ω. On this mesh we will build up the tensor product B-splines later. As mentioned in Section 3.2 we set Ω as a union of grid cells to preserve stability of B-splines and simplify implementation.

Definition 4. *Let ω be as before and G be a uniform grid with grid width $h > 0$ that covers ω. All cells $\sigma \subset G$ with*

$$\sigma \cap \omega \neq \emptyset$$

are called active cells. *The union of all active cells is called the* ambient domain Ω*, i.e.,*

$$\Omega := \bigcup \{\sigma \subset G \mid \sigma \text{ is an active cell}\}.$$

See Figure 4.2 for an example of active cells of a curve in $\mathbb{R}^2$.

Active cells are those cells that intersect the manifold ω. To get a well defined extension Ω should lie within an *embedded δ-tubular neighbourhood* Γ of ω. A *δ-tubular neighbourhood* for some $\delta > 0$ is a tube around ω with

$$\|x - y\|_2 < \delta \quad \text{for all } x \in \Gamma \text{ and } y \in \omega.$$

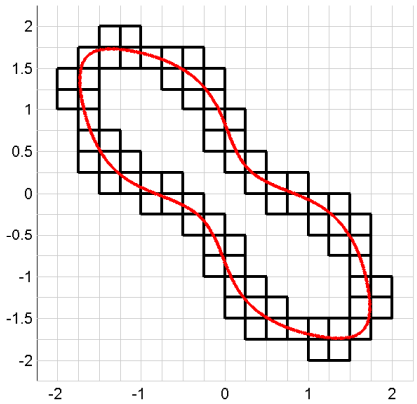

Figure 4.2: active cells of a curve in the plane

Figure 4.3: boundedness of the width of a tubular neighbourhood

It is called *embedded* if for each point $x \in \Gamma$ there exists exactly one point $y \in \omega$ such that for all points $z \in \omega \setminus \{y\}$

$$\|x - y\|_2 < \|x - z\|_2.$$

In the following, whenever we talk of a tubular neighbourhood we mean an embedded δ-tubular neighbourhood for some $\delta > 0$. The width δ is bounded by two criteria: On the one hand, it is bounded by the maximal curvature of ω (see Figure 4.3, left), i.e.,

$$\delta \leq \inf_{y \in \omega} \frac{1}{|\kappa|_{\max}(y)}$$

where $|\kappa|_{\max}(y)$ is the absolute value of the biggest normal curvature. On the other hand, δ is bounded by one half of the minimal distance of submanifold segments. See the right picture of Figure 4.3 for an example. Inside a tubular neighbourhood Γ the closest point function is well defined:

Definition 5. *Let ω be as above and Γ be an embedded δ-tubular neighbourhood of ω for some $\delta > 0$. The* closest point function $\mathrm{cp}\colon \Gamma \to \omega$ *is defined as*

$$\mathrm{cp}(x) := \operatorname*{argmin}_{y \in \omega} \|x - y\|_2 .$$

In [Foo84] it is shown that for smooth compact submanifolds the map cp is smooth as well. Moreover, the Tubular Neighbourhood Theorem states that for some $\delta > 0$ every point y in the δ-tubular neighbourhood can be uniquely described in the form $x + t\,\nu$ with $x \in \omega$, ν being the normal vector, and $t \in [-\delta, \delta]$.

Theorem 4 (Tubular Neighbourhood Theorem)**.** *Let ω be a compact smooth submanifold of codimension 1 embedded in $\mathbb{R}^d$. Then there exists $\delta > 0$ such that $\theta\colon \omega \times \mathbb{R} \to \{y \in \mathbb{R}^d \mid \exists x \in \omega$ with $\|y - x\|_2 < \delta\}$ with*

$$\theta(x, t) = x + t\,\nu(x) \quad \text{for } t \in [-\delta, \delta]$$

where $\nu(x)$ is the normal vector of ω at x, is a diffeomorphism.

The proof of this theorem is given in [Bre93].

Now, we choose δ such that the closest point function is well defined and the map θ of the Tubular Neighbourhood Theorem is a diffeomorphism and construct Ω such that it lies inside the δ-tubular neighbourhood. The diagonal of one cell is of length $h\sqrt{d}$ where d is the dimension. Therefore, let h be bounded by

$$0 < h < \frac{\delta}{\sqrt{d}}.$$

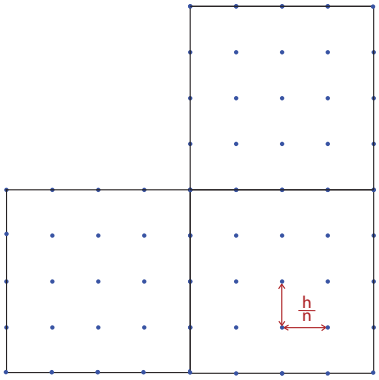

Figure 4.4: points inserted in each active cell

After computing the ambient domain Ω we select an order $n \in \mathbb{N}$. We use the same order in all coordinate directions, i.e., $n = n_1 = \cdots = n_d$. The choice of n depends on the application. According to the order n the resulting approximation will be in C^{n-2}. Yet, with n not only differentiability but also the computational time grows.

We want to extend the given function f to the ambient domain Ω where we will approximate it using tensor product B-splines of order n. Here, we need to consider only those B-splines whose supports include active cells.

Definition 6. *Let ω and n be as above. A B-spline b_i^n is called* active B-spline *if*

$$\operatorname{supp} b_i^n \cap \omega \neq \emptyset \quad i \in \mathbb{Z}^d$$

We can select any of the existing approximation methods on Ω. The Ambient B-spline Method works with the discrete local least squares approximation on Ω. The extended function F is evaluated at a finite set of discrete points $p_i \in \Omega$. Then the linear system of equations

$$B\xi = \bar{F}$$

has to be solved in the least squares sense, where $B_{ij} = b_j^n(p_i)$, ξ is the sought vector of coefficients and $\bar{F}_i = F(p_i)$ are the function values.

To do so we fill the active cells with data points to which we will assign function values. On each active cell exactly n^d B-splines are non-zero. There should be at least n^d data points per cell to avoid an underdetermined linear system of equations. We insert $(n+1)^d$ data points in each cell. The points are inserted as a fine regular grid of grid width $\frac{h}{n}$. Even on the boundary there are data points (see Figure 4.4). By inserting the data points in this way we avoid a scattered data problem, simplify the implementation (see Section 5.5), and fasten the computation. Moreover, we preserve stability of tensor product B-splines. We remark that the data points do not have to form a grid as long as there are at least n^d points per cell which are spread over the whole cell.

In order to assign values to the points we make use of the closest point function cp. The closest point function is either given or has to be calculated numerically. We assign to each point $p \in \Omega$ in the point grid the function value of $\operatorname{cp}(p) \in \omega$. The extension F then fulfils $F(p) = f(\operatorname{cp}(p))$. This results in a constant normal extension of f. In many applications we do not have a given function f on the submanifold but a finite set of discrete function values f_i. It may then happen that $f(\operatorname{cp}(p))$ is not given. In this case we first approximate $f(\operatorname{cp}(p))$ via linear interpolation.

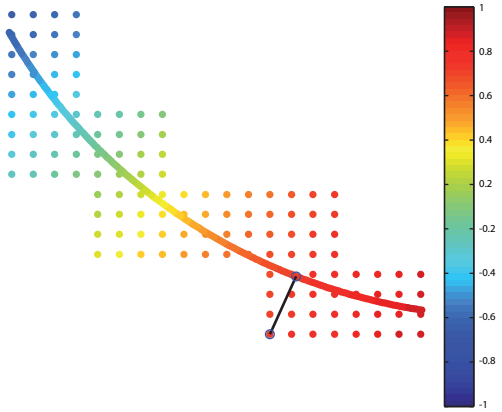

Figure 4.5: function values are assigned with the help of the closest point function [1]

Now, we approximate the function values by a spline $S \in S^n(\Omega)$. As basis, we have to consider all active B-splines. We use a variation of discrete local least squares approximation. That means we do not solve one linear system of equations for the whole problem but solve it locally. Let $I \subseteq \mathbb{Z}^d$ be the index set of all active B-splines and p_ℓ^i the data points in the support of b_i^n, $i \in I$. The discrete local least squares approximation solves

$$\sum_\ell \left(F(p_\ell^i) - S(p_\ell^i)\right)^2 \to \min \qquad \text{for all data points } p_\ell^i \in \operatorname{supp}(b_i^n).$$

For each $i \in I$ we obtain $(2n-1)^d$ coefficients of which only the coefficient of b_i^n is stored. The discrete local least squares approximation is a *quasi-interpolant of maximal order*.

Definition 7. *A* quasi-interpolant of order ν *is a linear operator* $Q\colon C^0 \to S^n$ *that is defined with the help of a family of functionals* $Q_i\colon C^0 \to \mathbb{R}$ *such that*

$$Qu := \sum_i (Q_i u) b_i^n .$$

A quasi-interpolant has to fulfil the following conditions:

- *The functionals* Q_i *are linear maps with* $Q_i u = Q_i u|_{[\tau_i, \tau_{i+n}]}$.
- *The quasi-interpolant reproduces polynomials of order* ν*, i.e.,* $Qp = p$ *for all* $p \in \mathbb{P}^\nu$.
- *The* Q_i *are uniformly bounded regarding the maximum norm.*

The following result on the approximation order of quasi-interpolants can be found in [HH13].

Theorem 5. *Let* f *be a smooth function and* Q *be a quasi-interpolant of maximal order* $n \in \mathbb{N}$*. Let* T *be a non-degenerated knot-sequence and* $x \in D(T)$*. Then the error satisfies*

$$|f(x) - Qf(x)| \le \frac{\|Q\|_\infty}{(n+1)!} \sum_{|\lambda|=n+1} \|f^{(\lambda)}\|_{L_\infty(D_x)} \prod_{i=1}^d h_i(x)^{\lambda_i},$$

[1] source: Nicole Lehmann, [Leh13]

where $f^{(\lambda)} = \frac{\partial^{|\lambda|} f}{\partial x_1^{\lambda_1} \dots \partial x_d^{\lambda_d}}$, D_x *is the union of the supports of all B-splines that do not vanish at* x, *and* $h_i(x) = \max_{y \in D_x} |x_i - y_i|$. *If the local mesh ratio is bounded by some number* r_T, *then*

$$|f^{(\mu)}(x) - Qf^{(\mu)}(x)| \leq c(n, r_T) \, \|Q\|_\infty \sum_{|\lambda|=n+1} \|f^{(\lambda)}\|_{L_\infty(D_x)} \prod_{i=1}^{d} h_i(x)^{\lambda_i - \mu_i}, , \quad |\mu| \leq n.$$

There are $(2n-1)^d$ B-splines b_j^n with $\operatorname{supp}(b_j^n) \cap \operatorname{supp}(b_k^n) \neq \emptyset$. It can happen that only few of these B-splines are active and that only one of the cells in $\operatorname{supp}(b_j^n)$ is filled with data points. Therefore, we modify the local least squares approximation without changing its order: Let σ be an active cell. We now take all n^d B-splines that do not vanish on σ. Each active cell σ is filled with $(n+1)^d$ data points. So, we formulate the local problem

$$\sum_i \big(F(p_i) - S(p_i)\big)^2 \to \min \qquad \text{for all data points } p_i \in \sigma \, .$$

This equals the linear system of equations

$$B^\top B \, \xi_\sigma = B^\top \bar{F}, \quad \text{with } B \in \mathbb{R}^{(n+1)^d \times n^d}, \; \xi_\sigma \in \mathbb{R}^{n^d} \text{ and } \bar{F} \in \mathbb{R}^{(n+1)^d} \tag{4.1}$$

where $B_{ij} = b_j^n(p_i), j \in J$ with J being the index set of the active B-splines, ξ_σ is the sought vector of coefficients and $\bar{F}_i = F(p_i)$. In contrast to the standard local least squares approximation we save all n^d coefficients. Solving this for each active cell, we obtain up to n^d coefficients for each B-spline. We take the average of them to achieve the desired coefficient. We obtain a spline $S(t) = \sum_j \xi_j \, b_j^n(t)$ for $t \in \Omega$ on the ambient domain Ω. Finally, we restrict the spline S to obtain a spline $s = S|_\omega$ on the submanifold ω. Therefore, we evaluate S on points $t \in \omega$,

$$s(t) = S(t) = \sum_j \xi_j \, b_j^n(t) \quad \text{for } t \in \omega.$$

If we want to approximate a multidimensional function $f : \omega \to \mathbb{R}^m$ with $m > 1$ each coordinate is treated separately. In that way the Ambient B-spline Method can be used to find smooth approximations of surfaces. Therefore a matching reference manifold of the same genus has to be chosen. A detailed description and numerical examples can be found in [Leh13].

4.2 Error Analysis

In this section we will give a theoretical proof of the approximation error that arises in the Ambient B-spline Method. We limit our analysis to submanifolds of codimension one. The workflow of the Ambient B-spline Method can be translated to submanifolds of arbitrary codimension. Yet, for codimension higher than one the theoretical proof presented in this section fails, as shown in Section 4.5.

4.2.1 Notations and Preliminaries

The Ambient B-spline Method does neither require any parametrisation of the submanifold nor any charts. Yet, the error analysis does.

Let ω be a smooth, compact submanifold of codimension one embedded in $\mathbb{R}^d$. Then, there exists a finite covering of open sets $\omega = \bigcup_{i \in I} \omega_i$ and a finite set of charts (υ_i, φ_i), where I is a finite index set, $\upsilon_i \subseteq \mathbb{R}^{d-1}$ and $\varphi_i \colon \upsilon_i \to \omega_i$ are n-times continuously differentiable homeomorphisms.

We define Sobolev-norms on manifolds as follows:

Definition 8. *Let ω and I be as above. For $1 \leq p \leq \infty$ and $f : \omega \to \mathbb{R}$* the Sobolev-norm on ω w.r.t. the atlas (v_i, φ_i), $i \in I$ *is given as*

$$\|f\|_{W_p^n(\omega)} := \begin{cases} \left(\sum_{i \in I} \|f \circ \varphi_i\|^p_{W_p^n(v_i)} \right)^{\frac{1}{p}} & \text{for } p < \infty \\ \max_{i \in I} \|f \circ \varphi_i\|_{W_\infty^n(v_i)} & \text{for } p = \infty \end{cases}$$

Then

$$W_p^n(\omega) := \{f : \omega \to \mathbb{R} \mid \|f\|_{W_p^n(\omega)} < \infty\}$$

defines a Sobolev space on ω.

The norm $\|\cdot\|_{W_p^n(\omega)}$ depends on the atlas. Yet, norms w.r.t. different atlases are equivalent. For the further analysis we suppose that an atlas is given. In [Dri03] Driver suggests another definition for Sobolev-norms on manifolds.

If for each $i \in I$ the norm $\|f\|_{W_p^n(\omega_i)}$ is bounded then so is $\|f\|_{W_p^n(\omega)}$. Therefore, to analyse the approximation error of the Ambient B-spline Method it suffices to consider only one such set ω_i. In the following we restrict our analysis to the closure of one set ω_i. That means we concentrate on a subset of the submanifold that is the closure of the image of a single chart (v_i, φ_i). For simplicity we omit the index i from now on. Whenever we talk about the submanifold ω we mean the closure of this subset.

Let ω be the submanifold and $\tilde{\Omega}$ be a tubular neighbourhood of ω with width $\varepsilon > 0$:

$$\tilde{\Omega} := \{x + t\nu(x) \mid x \in \omega,\ t \in [-\varepsilon, \varepsilon]\} \tag{4.2}$$

where $\nu(x)$ is the normal vector at $x \in \omega$. Here, ε has to be so small such that the map of Theorem 4 is a diffeomorphism (see Section 4.1).

Let $\varphi : v \to \omega$ be a local regular smooth parametrisation and $V := v \times [-\varepsilon, \varepsilon]$. We then define $\Phi : V \to \tilde{\Omega}$ as

$$\Phi(z, t) = \varphi(z) + t \cdot \nu(\varphi(z)) \quad \text{with } z \in v, t \in [-\varepsilon, \varepsilon].$$

Figure 4.6 visualises the current situation for a better understanding. Since φ is bijective we have by Theorem 4:

Corollary 2. *Let $\omega, \tilde{\Omega}, V, v$, and ε be as before. For a local smooth regular parametrisation $\varphi : v \to \omega$ the map*

$$\Phi : V \to \tilde{\Omega} \quad \textit{with} \quad \Phi(z, t) = \varphi(z) + t \cdot \nu(\varphi(z))$$

for $z \in v$ and $-\varepsilon \leq t \leq \varepsilon$ is a diffeomorphism.

Given a function $f : \omega \to \mathbb{R}$ we denote by $\hat{f}$ the corresponding function on the parameter space v, that means

$$\hat{f} : v \to \mathbb{R}, \qquad \hat{f} = f \circ \varphi\,.$$

Let E define an extension operator that extends functions defined on submanifolds of codimension one constantly in normal direction. We denote the extended functions by capital letters, i.e.,

$$F := Ef.$$

Then the following relationships hold:

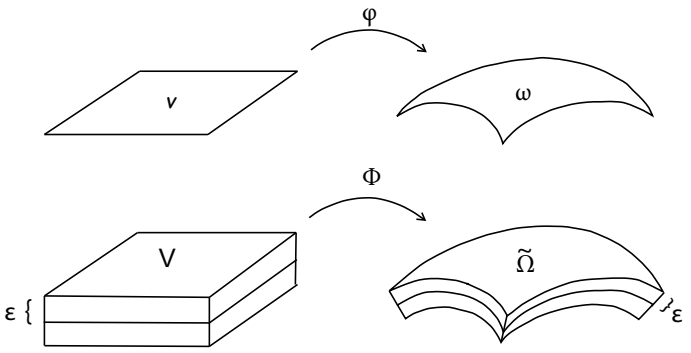

Figure 4.6: notations

- $\hat{F}: V \to \mathbb{R}, \qquad \hat{F} := E\hat{f}$
- $\hat{F} = F \circ \Phi$
- $F(x) = Ef(x) = f(\mathrm{cp}(x)) \quad$ for $x \in \Omega$.

In the main proof we use two known formulas. For completeness we give them here. The first one is the Faà di Bruno formula that is a generalisation of the chain rule to higher derivatives. We use the multivariate version of the Faà di Bruno formula as presented in [CS96]. First, we introduce some notations: Let $\alpha = (\alpha_1, \ldots, \alpha_m)$ for $m \in \mathbb{N}$ be a multi-index. Then

$$\alpha! := \prod_{i=1}^{m} (\alpha_i!) \quad \text{and}$$
$$[z]^\alpha := \prod_{i=1}^{m} z_i^{\alpha_i} \quad \text{for } z \in \mathbb{R}^m .$$

Moreover, we introduce the linear order $\prec$. For $\ell, \tilde{\ell} \in \mathbb{R}^m$ we write $\ell \prec \tilde{\ell}$ for either

$$|\ell| < |\tilde{\ell}| \quad \text{or}$$
$$|\ell| = |\tilde{\ell}|,\ \ell_1 < \tilde{\ell}_1 \quad \text{or}$$
$$|\ell| = |\tilde{\ell}|,\ \ell_1 = \tilde{\ell}_1,\ \ldots,\ \ell_k = \tilde{\ell}_k \text{ and } \ell_{k+1} < \tilde{\ell}_{k+1} \text{ for some } 1 \le k < m.$$

Let $x_0 \in \mathbb{R}^m$ and $w : \mathbb{R}^m \to \mathbb{R}^{\tilde{m}}$ with $w_1, \ldots, w_{\tilde{m}} \in C^\alpha(x_0)$. Let $y_0 \in \mathbb{R}^{\tilde{m}}$ with $y_0 = w(x_0)$. Now, let $u: \mathbb{R}^{\tilde{m}} \to \mathbb{R}$ with $u \in C^n(y_0)$ and let $h = u \circ w$ be the composition.

Theorem 6 (Faà di Bruno)**.** *Let x_0, y_0, u, w, and h be as above. Let α be an m-dimensional multi-index with $|\alpha| = n$. Then with $D_x^\alpha := \frac{\partial^{|\alpha|}}{\partial_{x_1}^{\alpha_1} \ldots \partial_{x_m}^{\alpha_m}}$ we have*

$$D_x^\alpha(h(x_0)) = \sum_{1 \le |\lambda| \le n} (D_y^\lambda u)(y_0) \cdot \sum_{s=1}^{n} \sum_{\substack{(k_1,\ldots,k_s,\ell_1,\ldots,\ell_s) \\ \in p_s(\alpha,\lambda)}} (\alpha!) \prod_{j=1}^{s} \frac{[G_{\ell_j}]^{k_j}}{(k_j!)(\ell_j!)^{|k_j|}}$$

where $G_{\ell_j} = (D_x^{\ell_j} w_1(x_0), \ldots, D_x^{\ell_j} w_{\tilde{m}}(x_0))$ and

$$p_s(\alpha, \lambda) = \left\{ (k_1, \ldots, k_s, \ell_1, \ldots, \ell_s) : |k_i| > 0,\ 0 \prec \ell_1 \prec \cdots \prec \ell_s,\ \sum_{i=1}^{s} k_i = \lambda,\ \sum_{i=1}^{s} |k_i| \ell_i = \alpha \right\} .$$

A proof of this theorem is given in [CS96]. It should be noted that the term

$$P_{\alpha,\lambda}(w) := \sum_{s=1}^{n} \sum_{\substack{(k_1,\dots,k_s,\ell_1,\dots,\ell_s) \\ \in p_s(\alpha,\lambda)}} (\alpha!) \prod_{j=1}^{s} \frac{[G_{\ell_j}]^{k_j}}{(k_j!)(\ell_j!)^{|k_j|}}$$

is a sum of polynomials in the partial derivatives of w. Here, the highest appearing derivative order of w is $D^\alpha w$. Therefore, we can combine these summands in a constant c depending on α and w.

The second theorem we use is Friedrichs' inequality. Friedrichs' inequality gives a bound of the L_p-norm of a function using its weak derivatives and the geometry of the domain.

Theorem 7 (Friedrichs' inequality)**.** *Let Ω be some domain contained in a d-dimensional cube of size h^d. Then*

$$\|f\|_{L_p} \leq h \left(\sum_{|\alpha|=1} \|\partial^\alpha f\|_{L_p}^p \right)^{\frac{1}{p}}$$

for all $f \in W_p^1(\Omega)$ with $f(x) = 0$ for all $x \in \partial\Omega$.

A proof of Friedrichs' inequality for $p = 2$ can for example be found in [Bra97]. For $p \neq 2$ we can adopt the main idea of the proof. A substitution of the Cauchy-Schwarz inequality by the Hölder inequality gives the result. We note that the condition of f vanishing at the boundary of Ω can also be substituted by the condition that f equals zero on a $(d-1)$-dimensional submanifold of Ω. The explicit value of the constant changes accordingly and becomes infinity when this submanifold shrinks to a $(d-2)$-dimensional submanifold.

4.2.2 Error Estimate

Now, we prove that the Ambient B-spline Method shows optimal error behaviour. The proof consists of two steps: First, we analyse the error behaviour for a tubular neighbourhood $\tilde{\Omega}$ as in (4.2). We state that given an approximation method of full approximation order on $\tilde{\Omega}$, we obtain optimal error behaviour on the submanifold as well. Second, we fit the Ambient B-spline Method, as described in Section 4.1, into this general case. We start with a lemma that verifies the norm equivalence of functions on V and $\tilde{\Omega}$.

Lemma 3. *Let $\tilde{\Omega}$, V, and Φ be as before. Let $1 \leq p \leq \infty$, $F \in W_p^n(\tilde{\Omega})$ and $\hat{F} = F \circ \Phi$. Then, $\hat{F} \in W_p^n(V)$ and there exist constants c, $C < \infty$ depending on n, p, and Φ such that*

$$c\,\|\hat{F}\|_{W_p^n(V)} \leq \|F\|_{W_p^n(\tilde{\Omega})} \leq C\,\|\hat{F}\|_{W_p^n(V)}\,.$$

Proof. Let α be a multi-index with $|\alpha| = n$. Applying the Faà di Bruno formula we get

$$\partial^\alpha \hat{F} = \partial^\alpha (F \circ \Phi) = \sum_{1 \leq |\lambda| \leq n} (\partial^\lambda F) \circ \Phi \cdot P_{\alpha,\lambda}(\Phi)\,,$$

where λ is a multi-index as well. The term $P_{\alpha,\lambda}(\Phi)$ is a polynomial of derivatives up to order n of Φ. On V this term is bounded. Throughout the following calculation we shall use c to denote a constant, not necessarily the same in any two places. We have

$$|\partial^\alpha \hat{F}| \leq c \left| \sum_{1 \leq |\lambda| \leq n} (\partial^\lambda F) \circ \Phi \right| \leq c \sum_{1 \leq |\lambda| \leq n} \left| (\partial^\lambda F) \circ \Phi \right|\,.$$

Let us first consider the case $p = \infty$. We have

$$\operatorname{ess\,sup} |\partial^\alpha \hat{F}| \leq c \operatorname{ess\,sup} \left| \sum_{1 \leq |\lambda| \leq n} (\partial^\lambda F) \circ \Phi \right| = c \operatorname{ess\,sup} \left| \sum_{1 \leq |\lambda| \leq n} \partial^\lambda F \right| \leq c \max_{|\lambda| \leq n} \|\partial^\lambda F\|_{L_\infty}.$$

From this it follows $\|\hat{F}\|_{W^n_\infty(V)} \leq c \|F\|_{W^n_\infty(\tilde{\Omega})}$. Now, let $1 \leq p < \infty$. Then it follows by norm equivalence that

$$|\partial^\alpha \hat{F}|^p \leq c \left(\sum_{1 \leq |\lambda| \leq n} \left| (\partial^\lambda F) \circ \Phi \right| \right)^p \leq c \sum_{1 \leq |\lambda| \leq n} \left| (\partial^\lambda F) \circ \Phi \right|^p .$$

Let J_Φ be the Jacoby matrix of Φ and $\det(J_\Phi)$ its determinant. It follows with Corollary 2 and the transformation rule that

$$\begin{aligned} \int_V |\partial^\alpha \hat{F}|^p &\leq c \int_V \sum_{1 \leq |\lambda| \leq n} |(\partial^\lambda F) \circ \Phi|^p \leq c \sum_{1 \leq |\lambda| \leq n} \frac{1}{\min |\det(J_\Phi)|} \int_V |(\partial^\lambda F) \circ \Phi|^p \cdot |\det(J_\Phi)| \\ &= c \int_{\tilde{\Omega}} \sum_{1 \leq |\lambda| \leq n} |(\partial^\lambda F)|^p \leq c \ \|F\|^p_{W^n_p(\tilde{\Omega})} \end{aligned}$$

This implies $\|\hat{F}\|_{W^n_p(V)} \leq c \|F\|_{W^n_p(\tilde{\Omega})}$.

Analogously, with $F = \hat{F} \circ \Phi^{-1}$ we obtain $\|F\|_{W^n_p(\tilde{\Omega})} \leq C \|\hat{F}\|_{W^n_p(V)}$ for some constant C that depends on the derivatives of Φ^{-1} up to order n. □

With this lemma we have proven the norm equivalence of $\|\cdot\|_{W^n_p(V)}$ and $\|\cdot\|_{W^n_p(\tilde{\Omega})}$. Now, we analyse how the extension operator E behaves under function norms. We recall that the operator E on a function $\hat{f} : v \to \mathbb{R}$ extends it to V in such a way that

$$E\hat{f} : V \to \mathbb{R}, \quad E\hat{f}(x,z) = \hat{f}(x), \quad \text{with } x \in v,\ z \in \mathbb{R}.$$

Lemma 4. *Let v, V, and E be as defined before. Let $\hat{f} \in W^n_p(v)$ and $\hat{F} := E\hat{f}$. Then, $\hat{F} \in W^n_p(V)$ and for $1 \leq p < \infty$*

$$\|\hat{F}\|_{W^n_p(V)} = (2\varepsilon)^{\frac{1}{p}} \|\hat{f}\|_{W^n_p(v)}.$$

For $p = \infty$ we have

$$\|\hat{F}\|_{W^n_\infty(V)} = \|\hat{f}\|_{W^n_\infty(v)}.$$

Proof. Let Λ be a multi-index of size d and λ a multi-index of size $d-1$ with $\lambda = [\Lambda_1, \ldots, \Lambda_{d-1}]$. We first consider the case $p < \infty$. Keeping in mind that $\hat{F}$ is constant in the direction of the last component we obtain

$$\begin{aligned} \|\hat{F}\|^p_{W^n_p(V)} &= \sum_{|\Lambda| \leq n} \|\partial^\Lambda \hat{F}\|^p_{L_p(V)} = \sum_{|\lambda| \leq n} \|\partial^{(\lambda,0)} \hat{F}\|^p_{L_p(V)} \\ &= \sum_{|\lambda| \leq n} \int_v \int_{-\varepsilon}^{\varepsilon} |\partial^{(\lambda,0)} \hat{F}(x,z)|^p \, dz \, dx = \sum_{|\lambda| \leq n} 2\varepsilon \int_v |\partial^\lambda \hat{f}(x)|^p \, dx = 2\varepsilon \, \|\hat{f}\|^p_{W^n_p(v)}. \end{aligned}$$

If $p = \infty$ then we have

$$\|\hat{F}\|_{W^n_\infty(V)} = \sum_{|\Lambda| \leq n} \|\partial^\Lambda \hat{F}\|_{L_\infty(V)} = \sum_{|\lambda| \leq n} \operatorname*{ess\,sup}_{x \in V} \left(|\partial^{(\lambda,0)} \hat{F}| \right) = \sum_{|\lambda| \leq n} \operatorname*{ess\,sup}_{x \in v} \left(|\partial^\lambda \hat{f}| \right) = \|\hat{f}\|_{W^n_\infty(v)}.$$

□

We will now show that we have optimal error behaviour on the submanifold provided we have an approximation method with optimal approximation order on $\tilde{\Omega}$.

Theorem 8. *Let ω and E be as before, $1 \leq p \leq \infty$, and $f \in W_p^n(\omega)$. Let $\tilde{\Omega}$ be as before and $h > 0$ be a discretising parameter. Let $\mathscr{M}_h$ be an approximation method for functions $F\colon \tilde{\Omega} \to \mathbb{R}$ such that $S := \mathscr{M}_h(F)$ fulfils*

$$\|S - F\|_{W_p^m(\tilde{\Omega})} \leq Ch^{n-m}\|F\|_{W_p^n(\tilde{\Omega})}, \quad m < n,$$

for a constant C. Let

$$s := \left(\mathscr{M}_h(Ef)\right)|_\omega .$$

Then we have

$$\|f - s\|_{W_p^m(\omega)} \leq c\, h^{n-m}\|f\|_{W_p^n(\omega)}$$

for some constant c and $m < n - 1$.

Proof. Let φ, v, and V be as introduced before. We set $\varepsilon = Ch$ with some constant $0 < C < \frac{1}{\sqrt{h}}$ such that $\Phi\colon v \times [-\varepsilon, \varepsilon] \to \tilde{\Omega}$ is a diffeomorphism. By definition we have $\|s - f\|_{W_p^m(\omega)} = \|\hat{s} - \hat{f}\|_{W_p^m(v)}$. Let us first consider the case $p < \infty$. With Lemma 4 it follows that

$$\|\hat{s} - \hat{f}\|_{W_p^m(v)} = (C\,h)^{-\frac{1}{p}}\,\|E(\hat{s} - \hat{f})\|_{W_p^m(V)} = c\,h^{-\frac{1}{p}}\left(\sum_{|\mu|\leq m}\|\partial^{(\mu,0)}(E\hat{s} - \hat{F})\|_{L_p(V)}^p\right)^{\frac{1}{p}}$$

$$\leq c\,h^{-\frac{1}{p}}\left(\left(\sum_{|\mu|\leq m}\|\partial^{(\mu,0)}(E\hat{s} - \hat{S})\|_{L_p(V)}^p\right)^{\frac{1}{p}} + \left(\sum_{|\mu|\leq m}\|\partial^{(\mu,0)}(\hat{S} - \hat{F})\|_{L_p(V)}^p\right)^{\frac{1}{p}}\right) \tag{4.3}$$

where μ is a multi-index and $\hat{F} = E\hat{f}$. We consider both terms separately, starting with the second one. With the help of the above lemmata we obtain:

$$\left(\sum_{|\mu|\leq m}\|\partial^{(\mu,0)}(\hat{S} - \hat{F})\|_{L_p(V)}^p\right)^{\frac{1}{p}} \leq \|\hat{S} - \hat{F}\|_{W_p^m(V)} \leq c\,\|S - F\|_{W_p^m(\tilde{\Omega})}$$

$$\leq ch^{n-m}\|F\|_{W_p^n(\tilde{\Omega})} \leq ch^{n-m}\|\hat{F}\|_{W_p^n(V)} \leq ch^{n-m}h^{\frac{1}{p}}\,\|\hat{f}\|_{W_p^n(v)}. \tag{*}$$

Now, we will have a look at the first term. The function $(E\hat{s} - \hat{S})\colon V \to \mathbb{R}$ vanishes on v. We fix $x \in v$. Let $(E\hat{s} - \hat{S})_{x,\mu}\colon [-C\,h, C\,h] \to \mathbb{R}$ be given as

$$(E\hat{s} - \hat{S})_{x,\mu}(z) := \partial^{(\mu,0)}(E\hat{s} - \hat{S})(x,z)\,.$$

for a multi-index μ with $|\mu| \leq (n-2)$. Then for any $x \in v$ and any multi-index μ the function $(E\hat{s} - \hat{S})_{x,\mu}$ is a one dimensional function that vanishes at the centre, i.e.,

$$(E\hat{s} - \hat{S})_{x,\mu}(0) = 0\,.$$

Now, by Friedrichs' inequality:

$$\|(E\hat{s}-\hat{S})_{x,\mu}\|^p_{L_p([-C\,h,C\,h])} \le c\,h^p \|\partial(E\hat{s}-\hat{S})_{x,\mu}\|^p_{L_p([-C\,h,C\,h])}.$$

Therefore, for $m < (n-1)$

$$\left(\sum_{|\mu|\le m} \|\partial^{(\mu,0)}(E\hat{s}-\hat{S})\|^p_{L_p(V)}\right)^{\frac{1}{p}} \le c\,h \left(\sum_{|\mu|\le m} \|\partial^{(\mu,1)}(E\hat{s}-\hat{S})\|^p_{L_p(V)}\right)^{\frac{1}{p}}$$

$$= c\,h \left(\sum_{|\mu|\le m} \|\partial^{(\mu,1)}(\hat{S}-\hat{F})\|^p_{L_p(V)}\right)^{\frac{1}{p}} \le c\,h\,\|\hat{S}-\hat{F}\|_{W^{m+1}_p(V)} \le c\,h\,\|S-F\|_{W^{m+1}_p(\tilde{\Omega})}$$

$$\le c\,h^{n-m}\|F\|_{W^n_p(\tilde{\Omega})} \le ch^{n-m}h^{\frac{1}{p}}\|f\|_{W^n_p(\omega)}. \qquad (**)$$

Inserting (*) and (**) in (4.3) leads to the desired result.

In the case $p=\infty$ (4.3) simplifies to

$$\|\hat{s}-\hat{f}\|_{W^m_\infty(\nu)} \le \max_{|\mu|\le m} \|\partial^{(\mu,0)}(E\hat{s}-\hat{S})\|_{L_\infty(V)} + \max_{|\mu|\le m} \|\partial^{(\mu,0)}(\hat{S}-\hat{F})\|_{L_\infty(V)}. \qquad (4.3\ ')$$

Again we consider both terms separately. For the second term we have with the same arguments as above

$$\max_{|\mu|\le m} \|\partial^{(\mu,0)}(\hat{S}-\hat{F})\|_{L_\infty(V)} \le \|\hat{S}-\hat{F}\|_{W^m_\infty(V)} \le ch^{n-m}\|\hat{F}\|_{W^n_\infty(V)} \le ch^{n-m}\|\hat{f}\|_{W^n_\infty(\nu)}.$$

For the first term we have

$$\max_{|\mu|\le m} \|\partial^{(\mu,0)}(E\hat{s}-\hat{S})\|_{L_\infty(V)} \le h \max_{|\mu|\le m} \|\partial^{(\mu,1)}(E\hat{s}-\hat{S})\|_{L_\infty(V)} \le h\,\|\hat{S}-\hat{F}\|_{W^{m+1}_\infty(V)} \le ch^{n-m}\|f\|_{W^n_\infty(\omega)}.$$

□

The existence of approximation methods $\mathscr{M}_h$ with full approximation order on arbitrary domains is shown in [DDS80].

We will now prove that the Ambient B-spline Method is of optimal error behaviour. In Theorem 8 we consider tubular neighbourhoods as in (4.2). Yet, the ambient domain of the Ambient B-spline Method is of different shape. Therefore, it remains to prove that we can adopt the idea of Theorem 8 to this case. First, we show that the ambient domain contains a tubular neighbourhood with width $\varepsilon \in \theta(h)$. Then we can limit the extension to that tubular neighbourhood without loosing approximation power and use Theorem 8 to prove optimal error behaviour of the Ambient B-spline Method.

Let Ω be the union of all active cells (see Section 4.1). Let $S\colon \Omega \to \mathbb{R}$ denote a tensor product B-spline approximation of $F \in W^n_p(\Omega)$ of order n on a grid with grid width h such that

$$\|S-F\|_{W^m_p(\Omega)} \le ch^{n-m}\|F\|_{W^n_p(\Omega)}, \quad m<n,$$

for a constant c independent of h. Until now we denoted by ω the closure of one set ω_i. In the following theorem we consider the whole submanifold ω.

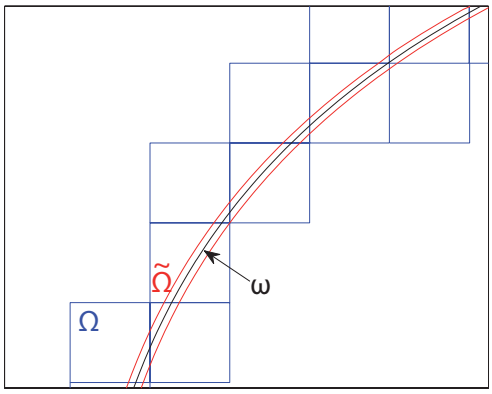

Figure 4.7: $\tilde{\Omega}$ lies inside Ω

Theorem 9. *Let ω be as before and $h_0 > 0$ be such that the closest point projection is well defined on the active cells of grid width h_0. Let $1 \leq p \leq \infty$ and $f \in W_p^n(\omega)$. Let s be the approximation of f obtained by using the Ambient B-spline Method. Then*

$$\|f - s\|_{W_p^m(\omega)} \leq c\, h^{n-m} \|f\|_{W_p^n(\omega)}, \quad m < n-1 \text{ and } h \leq h_0$$

for some constant c depending on n and p.

Proof. As before, Ω is the union of all active cells and let $\partial\Omega$ be its boundary. Now, we define

$$\delta := \inf_{X \in \partial\Omega} \{|X - \mathrm{cp}(X)|\}.$$

Since $\partial\Omega$ is compact we have $\delta > 0$. So we find a constant $C > 0$ such that $\delta = Ch$. Moreover we have $\delta < \sqrt{d}h$. We set

$$\tilde{\Omega} := \{x + tv(x) \,|\, x \in \omega, t \in [-\delta, \delta]\},$$

where v is the normal vector. Then, $\tilde{\Omega}$ fulfils the conditions of Theorem 8 with $\varepsilon = \delta$ and by construction $\tilde{\Omega} \subset \Omega$ (see Figure 4.7).

Let $\mathscr{M}_h$ be an approximation method that constructs the tensor product B-spline approximation on Ω and returns the restriction to $\tilde{\Omega}$ of this spline. So,

$$\mathscr{M}_h(F) = S|_{\tilde{\Omega}}.$$

Since $\tilde{\Omega}$ is a d-dimensional subset of $\Omega \subset \mathbb{R}^d$ the restriction $S|_{\tilde{\Omega}}$ fulfils

$$\|F|_{\tilde{\Omega}} - S|_{\tilde{\Omega}}\|_{W_p^m(\tilde{\Omega})} = \|(F-S)|_{\tilde{\Omega}}\|_{W_p^m(\tilde{\Omega})} \leq ch^{n-m}\|F\|_{W_p^n(\tilde{\Omega})}.$$

Then by Theorem 8

$$\bar{s} := (S|_{\tilde{\Omega}})|_{\omega}$$

fulfils

$$\|f - \bar{s}\|_{W_p^m(\omega)} \leq c\, h^{n-m} \|f\|_{W_p^n(\omega)}.$$

We have $\bar{s} = s$ which proves the theorem. □

4.3 Implementation

Let us now have a look at how each step of the Ambient B-spline Method is implemented. The implementation is done in MATLAB. The INPUT of the method is

- a manifold ω,
- a continuous function $f : \omega \to \mathbb{R}$,
- a grid width $h > 0$ and
- an order $n \in \mathbb{N}$.

The manifold is given implicitly. That means ω is a level set of a smooth function. See Section 4.4 for examples. The order n is not a vector but a number because we use only tensor product B-splines with the same order in each direction. If the function f is not given as a continuous function but as discrete values f_i or if f is multidimensional then small variations have to be added to the implementation.

The OUTPUT is the index set I of all active B-splines, a vector ξ containing the coefficients of the spline $S = \sum_{i \in I} \xi_i \, b_i^n$ and the resulting error which contains the maximal as well as the root mean square error.

First of all, we construct a uniform d-dimensional grid $\bigotimes_{i=1}^{d} T_i$ of grid size h that covers the whole manifold. For simplicity, the shift is set to zero which means that the origin is a cell corner. The next step is to compute the active cells. To make sure that no active cell is missed out by numerical inaccuracy all cells $\sigma \subset G$ with

$$\inf_{x \in \sigma} \{ \|x - y\|_2 \mid y \in \omega \} \leq \frac{h}{100}$$

are considered as active cells. Each cell and each B-spline is identified by the corner with the smallest indices. For example, on the two dimensional grid $T_1 \times T_2$ the cell $\sigma_{[3,4]}$ equals $\left[\tau_3^1, \tau_4^1\right] \times \left[\tau_4^2, \tau_5^2\right]$ and the support of the B-spline $b_{[3,4]}^3$ is

$$\operatorname{supp} b_{[3,4]}^3 = \left[\tau_3^1, \tau_6^1\right] \times \left[\tau_4^2, \tau_7^2\right] .$$

To find the active cells, we distribute a number of discrete points p_i on the submanifold. To each point we add and subtract $\frac{h}{100} \, \nu(p_i)$ where ν is the normal on the submanifold. All these points are rounded to their corresponding grid corner, i.e., the lower left corner of the cell in which the point lies. The cells that are identified with these corners are active cells.

Next, we insert the data points q_i into the active cells. Therefore, we construct a regular point grid

$$\bigotimes_{i=1}^{d} I_i \quad \text{with } I_i = [0, \tfrac{h}{n}, \ldots, h] \text{ for all } i = 1, \ldots, d.$$

Then we add this grid to the lower left corner of each active cell. On each active cell σ we have to solve (4.1) with

$$B_{i,j} = b_j^n(q_i) \quad \text{for points } q_i.$$

Since the grid is uniform, all B-splines are uniform B-splines. Moreover, the points are arranged in the same way in each active cell. Therefore, for each active cell the matrix B looks the same. So, we have to compute B only once. This saves calculation time as well as memory.

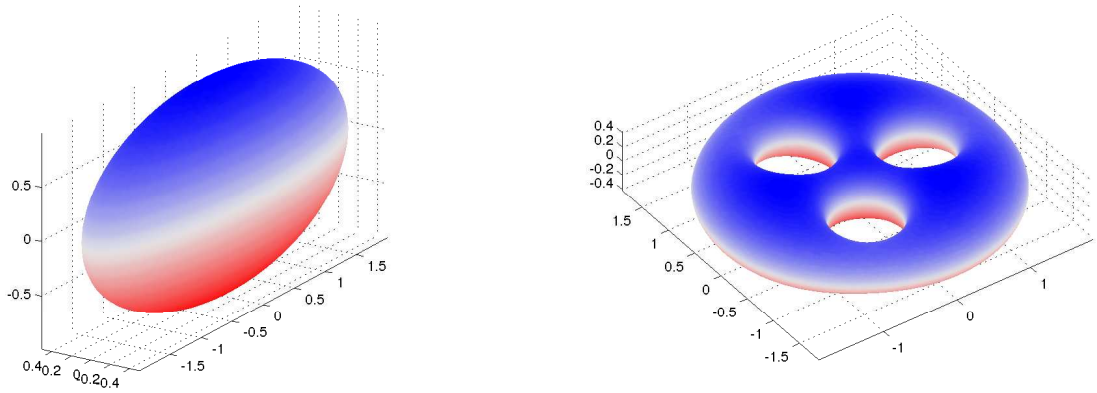

Figure 4.8: the test manifolds

Next, we compute the closest points on the submanifold of each of these points. If the closest point function is not explicitly given or easy to calculate, we have to compute the closest points numerically. This is in general numerically expensive. Several algorithms do already exist to find the closest point. See [HW05] for a short overview and a second order algorithm for curves and surfaces. We first make a guess by using the gradient of the level set function and then improve the guess by Newton iteration until the error is less than a given tolerance.

The next step is to assign function values to the data points. If the input function f is given as a continuous function on ω we can just evaluate f at the points $\mathrm{cp}(q_i)$ to obtain $F(q_i) := f(\mathrm{cp}(q_i))$. Otherwise we have to approximate it using linear interpolation.

Finally, we solve $(B\xi - F)^2 \to \min$ on each active cell by using the backslash operator in MATLAB. The backslash operator solves the linear equation $B^\top B\xi = B^\top F$. The obtained coefficients ξ are saved along with a counter. This counter returns for each active B-spline the number of active cells in its support. Finally, we add up the coefficients and compute the average value by dividing through the counter value.

To obtain the error of the resulting spline we evaluate the spline $s(x_\ell) = S(x_\ell) = B(x_\ell)\xi$ at a finite number of discrete points $x_\ell \in \omega$ and compare the values with $f(x_\ell)$. We compute the maximal error

$$\mathrm{err}_{\max} := \max_\ell |s(x_\ell) - f(x_\ell)|$$

as well as the root mean square error

$$\mathrm{err}_{\mathrm{rms}} := \left(\frac{\sum_\ell |s(x_\ell) - f(x_\ell)|^2}{\#x_\ell} \right)^{\frac{1}{2}},$$

where $\#x_\ell$ denotes the number of evaluation points.

4.4 Numerical Examples

In order to confirm the theoretically proven rate of convergence of Section 4.2 we apply the Ambient B-spline Method to smooth test functions on two implicitly given submanifolds that are embedded in $\mathbb{R}^3$.

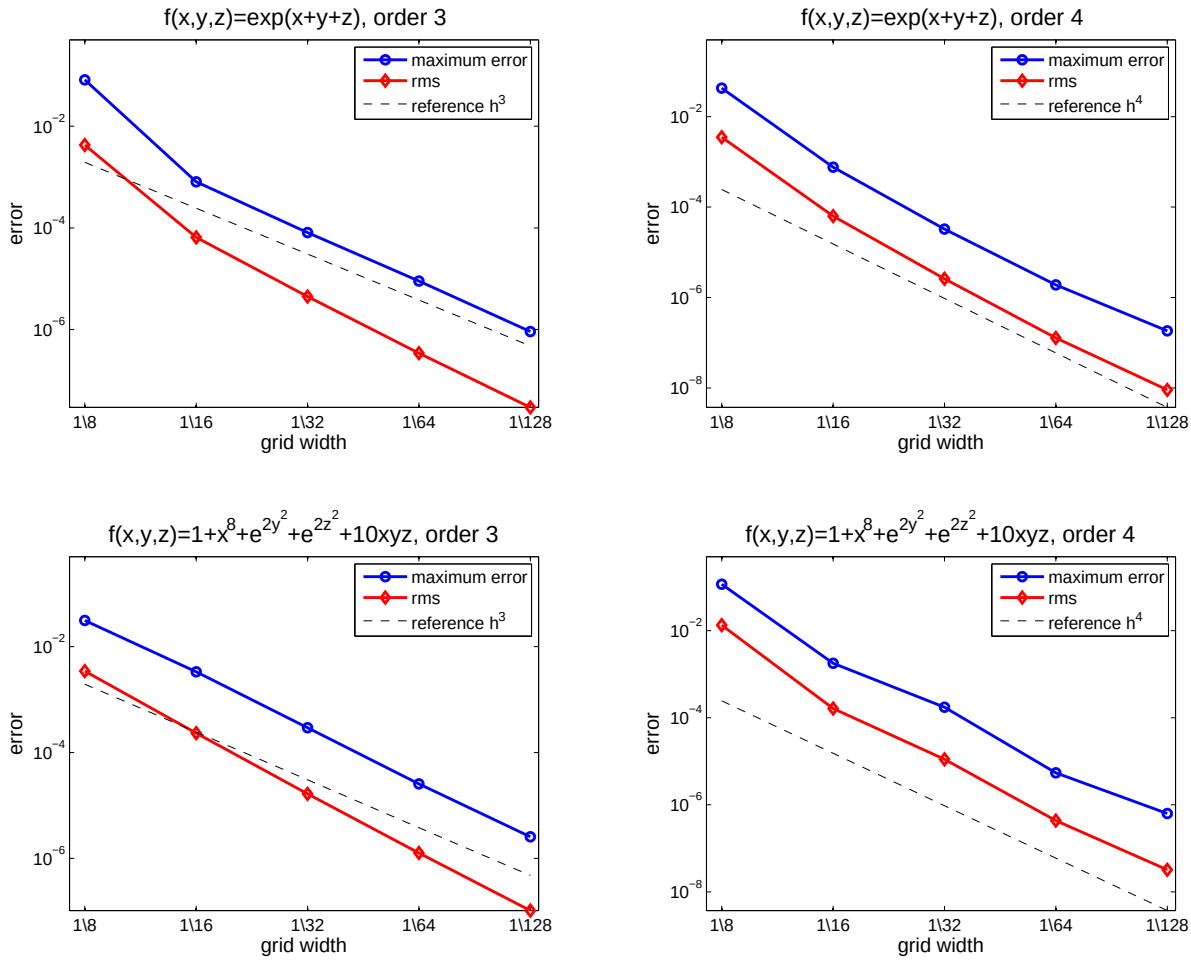

Figure 4.9: experimental rate of convergence on the ellipsoid

The first submanifold is the ellipsoid given as the zero set of

$$\varphi_1(x,y,z) = x^2 + 16y^2 + 4z^2 - 4\,.$$

The second submanifold is a genus three manifold with holes centred around

$$x_k = \frac{3}{4}\,\mathrm{Re}(e^{-\frac{2i\pi k}{3}}), \quad y_k = \frac{3}{4}\,\mathrm{Im}(e^{-\frac{2i\pi k}{3}}), \quad z_k = 0 \quad \text{for } k = 0,1,2\,.$$

It is given as zero set of

$$\varphi_2(x,y,z) = x^2 + y^2 + 9z^2 + \frac{1}{2}\sum_{i=0}^{2}((x-x_i)^2 + (y-y_i)^2)^{-1} - 4\,.$$

Both test manifolds are depicted in Figure 4.8.

The Ambient B-spline Method is used as follows: A function f is chosen and the Ambient B-spline Method is used as described in the previous sections to find a smooth approximation $s \in S^n$ of $f \in W_p^n(\omega)$ on the submanifold. The method is implemented as described in Section 4.3 in MATLAB. We evaluate the resulting splines at nearly 600000 equally distributed points x_ℓ, $\ell = 1,\ldots,600000$

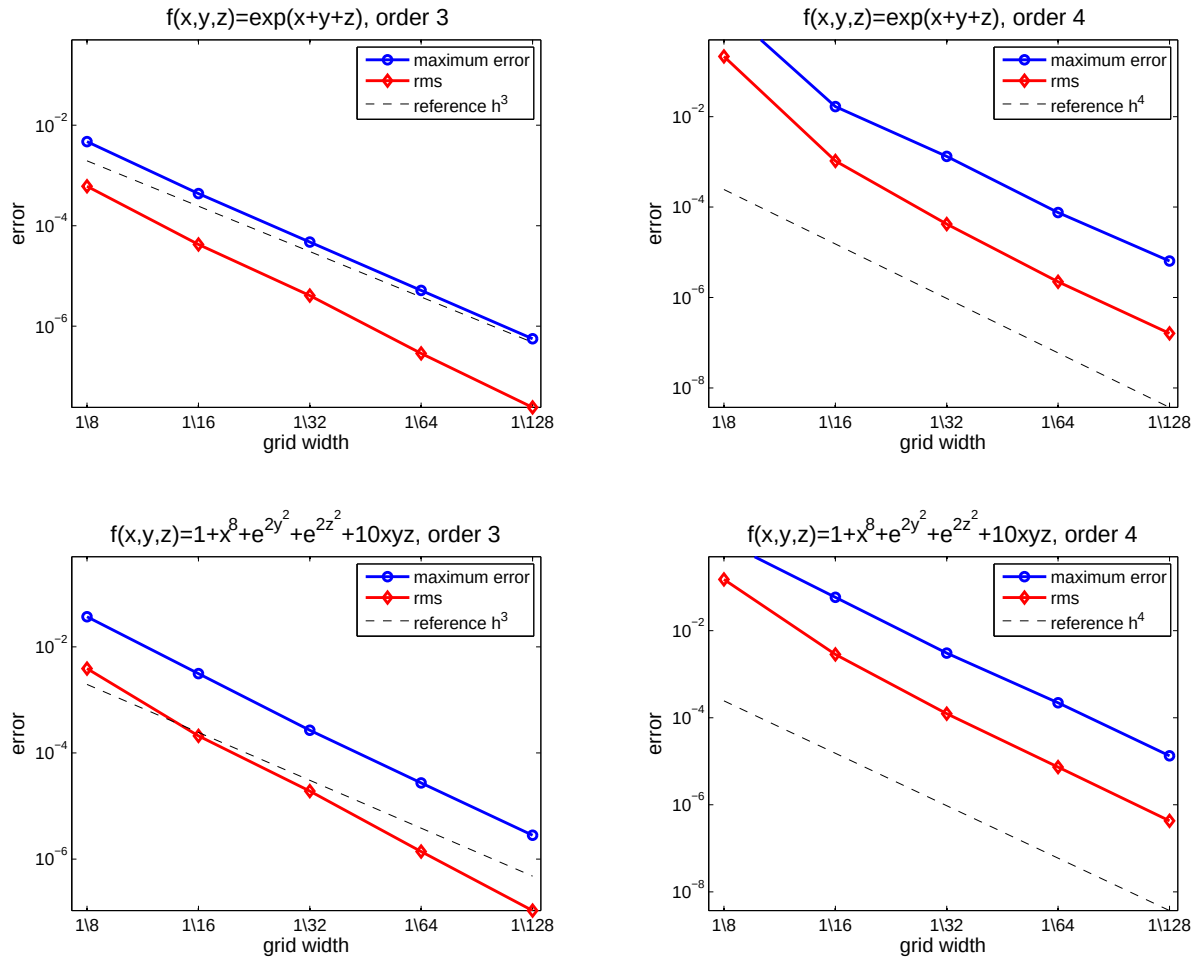

Figure 4.10: experimental rate of convergence on the genus three manifold

on the manifold and compare them with the exact function values. As before mentioned two kinds of errors are computed: the maximum error and the root mean square error.

Approximations were computed for grid widths ranging from $\frac{1}{8}$ to $\frac{1}{128}$ to analyse convergence. Figure 4.9 shows the error on the ellipsoid for two test functions. The test functions used are

$$f_1(x,y,z) = \exp(x+y+z) \quad \text{and} \quad f_2(x,y,z) := 1 + x^8 + \exp(2y^2) + \exp(2z^2) + 10xyz\,.$$

For both test functions the maximum error (blue) and the root mean squares error (red) are calculated and depicted for the orders $n = 3$ and $n = 4$. That means we construct a C^1 and a C^2-approximation. The dashed line is a reference line representing h^n. Both axes have a logarithmic scale. In all four diagrams both errors achieve the expected rate of convergence.

For the second test manifold we choose the same approach. Again the grid width h ranges from $\frac{1}{8}$ to $\frac{1}{128}$ and the orders are $n = 3$ and $n = 4$. We apply the Ambient B-spline Method on the same test functions f_1 and f_2 as before and calculate the maximal as well as the root mean square error. Figure 4.10 depicts the approximation errors on the second test manifold. We again added the reference line h^n for a better comparison. As we can see the expected rate of convergence is achieved for this manifold as well.

4.5 Higher Codimension

Until now we considered only submanifolds of codimension one. The workflow of the Ambient B-spline Method is transferable to submanifolds of any codimension. Unfortunately, the proof for the error convergence of the method as presented in Section 4.2 is not valid for submanifolds of codimension higher than one. In this section we will have a closer look why the proof fails and present some considerations under which circumstances convergence is still given.

The workflow of the method remains unchanged in any codimension: We find all active cells and fill them with a fine regular point grid. Then, function values are assigned to these data points according to their closest point on the submanifold. Finally, the points are approximated by tensor product B-splines of the according dimension. Figure 4.11 shows an example for a curve embedded in $\mathbb{R}^3$.

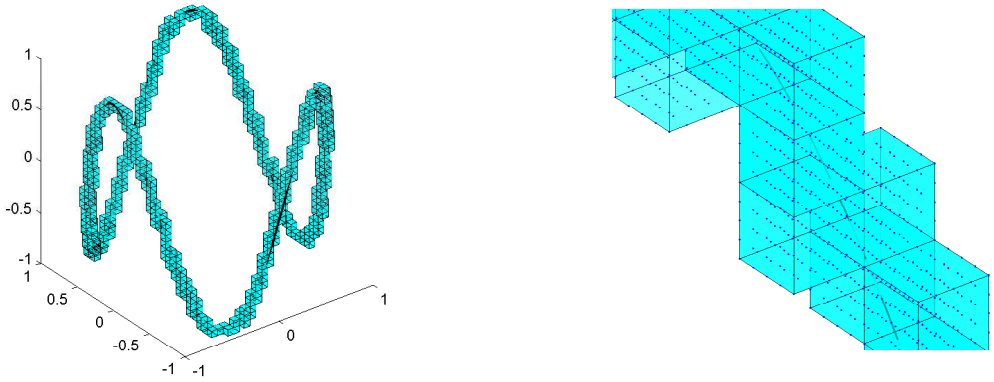

Figure 4.11: *left:* active cells for a curve in $\mathbb{R}^3$, *right:* active cells filled with data points

Let us now have a look at the error analysis. We consider a submanifold ω of dimension k in $\mathbb{R}^d$, i.e., the submanifold is of codimension $d-k$. We keep the notations of Section 4.2. Then, Lemma 3 and Lemma 4 transfer into the case of higher codimension without any complications. Now, for the main proof we need to estimate $E\hat{s}-\hat{S}$ in terms of its gradient. For codimension one this was given by Friedrichs' inequality.

Let us have a closer look at the function $E\hat{s}-\hat{S}: V \to \mathbb{R}$ with $V = \nu \times [-\varepsilon,\varepsilon]^{d-k}$ where ν is a parameter domain of ω (see the notations introduced in Section 4.2). We note that $\hat{S}$ is a spline and $E\hat{s}$ is the constant extension of $\hat{s} = \hat{S}|_\nu$. Moreover, $(E\hat{s}-\hat{S})(x,\mathbf{0}) = 0$ for all $x \in \nu$ and $\mathbf{0}$ be the $(d-k)$-dimensional zero vector. Now, we need an estimation of the form

$$\|\partial^{(\mu,\mathbf{0})}(E\hat{s}-\hat{S})\|_{L_p(V)} \leq c\,h\,\|\partial^{(\mu,\lambda)}(E\hat{s}-\hat{S})\|_{L_p(V)}$$

where $\mu \in \mathbb{R}^k$ is a multi-index, $\mathbf{0}$ is a $(d-k)$-dimensonal zero vector and $\lambda \in \mathbb{R}^{d-k}$ is a multi-index with $|\lambda| = 1$.

Let $x \in \nu$ be fixed. Then, for any multi-index $\mu \in \mathbb{R}^k$ the function $\partial^{(\mu,0)}(E\hat{s}-\hat{S})(x,z)$ is zero for $z = \mathbf{0}$. This is a 0-dimensional subspace of a $(d-k)$-dimensional space. Friedrichs' inequality can be applied only if the difference of the space dimensions is one. Therefore, for $d-k > 1$ we cannot apply Friedrichs' inequality to obtain the desired estimation.

However, concerning the maximal norm the Ambient B-spline Method is of optimal approximation power for submanifolds of arbitrary codimension.

Theorem 10. *Let ω be a smooth compact submanifold embedded in $\mathbb{R}^d$. Let s, S and h be defined as in Theorem 9. Then*

$$\|f - s\|_{W_\infty^m(\omega)} \leq c\, h^{n-m} \|f\|_{W_\infty^n(\omega)}$$

where c is a constant depending on n.

Proof. Let E be the extension operator with $Ef(x_1, \ldots, x_d) = f(x_1, \ldots, x_k)$. We note that

$$|\partial^{(\mu,0)}(E\hat{s} - \hat{S})(x,z)| = |\partial^{(\mu,0)}(E\hat{s} - \hat{S})(x,z) - \partial^{(\mu,0)}(E\hat{s} - \hat{S})(x,0)| \leq c\, h \max_{|\lambda|=1} |\partial^{(\mu,\lambda)}(E\hat{s} - \hat{S})(x,\zeta)|$$

for a constant c and some $\zeta \in [-\varepsilon, \varepsilon]^{d-k}$ and therefore we have

$$\|\partial^{(\mu,0)}(E\hat{s} - \hat{S})\|_{L_\infty(V)} \leq c\, h \|\partial^{(\mu,\lambda)}(E\hat{s} - \hat{S})\|_{L_\infty(V)}.$$

This means that for $p = \infty$ Friedrichs' inequality can also be applied to submanifolds of arbitrary codimension. Then the proof of Theorem 9 can be transfered into the case of higher codimension without any complications. □

5 Ambient Signed Distance Method

PDEs on manifolds is a field that gains more and more interest in different areas of application, like for example geophysics, image processing and medicine. Efficient, fast and easy to implement methods are required to solve these kinds of problems.

In this chapter we present an embedding method to solve intrinsic linear elliptic second-order PDEs on submanifolds. The workflow is similar to the concept of the Ambient B-spline Method, see Chapter 4: Given a problem on a compact submanifold embedded in $\mathbb{R}^d$ we extend it to an ambient domain. There, we solve the problem with already known techniques and finally restrict the result to the submanifold.

Compared to the Ambient B-spline Method some new aspects have to be added. The given PDE is an intrinsic PDE given on the submanifold. Obviously, it has to be modified so that it becomes a PDE on some ambient domain. We call this modified PDE *embedding PDE*. The construction of this embedding PDE represents an important part of the method. The original intrinsic PDE needs to be translated into an embedding PDE such that both solutions coincide on the submanifold. Operators in the given intrinsic PDE, like the intrinsic gradient or the Laplace-Beltrami operator, have to be modified. Here, we make use of the signed distance function. This gives the method its name. We construct the embedding PDE in such a way that the solution is constant in normal direction of the submanifold. Basic ideas of [DE13] and [RM08] are included in this method to find the modification.

Since the embedding PDE is then given on a subset of $\mathbb{R}^d$, boundary conditions have to be formulated. Due to the structure of the domain we cannot draw on existing standard boundary conditions. Instead, we present a new formulation of boundary conditions. We "thicken" the boundary such that it becomes a d-dimensional subset of $\mathbb{R}^d$. Then, we solve two equations: the embedding PDE near the submanifold and the boundary condition at the "thickened" boundary.

Finally, the embedding PDE is solved using the Finite Element Method. As in the Ambient B-spline Method we restrict the given problem to the spline space S^n with tensor product B-splines as basis functions. The FEM leads to a linear system of equations. We solve this for the control points of the spline that approximates the solution of the embedding PDE. We restrict this spline to the submanifold and result in an approximation of the solution of the given intrinsic PDE.

The main concept of the method works for any kind of smooth compact submanifold and any linear second-order PDE given on it. Yet, we restrict our analysis to compact submanifolds of codimension one and static elliptic PDEs. We concentrate on the model equation $-\triangle_\omega u + cu = f$ for a constant $c > 0$ and a function f given on the submanifold.

The Ambient Signed Distance Method returns arbitrarily smooth approximations of solutions of linear intrinsic PDEs. It works on any smooth compact submanifold of codimension one, independently of the genus or the dimension. The method has a simple workflow, is easy to implement and can be generalised to parabolic PDEs as well. No triangulation or parametrisation of the manifold is needed and we don't have to construct complicated function spaces on the manifold itself. Moreover, if the given PDE is elliptic then so is the embedding PDE. This is the main advantage that distinguishes the method from a similar approach by Dzuik and Elliott.

First, we introduce some preliminaries and notations in Section 5.1. Section 5.2 shows how the embedding PDE is constructed while Section 5.3 deals with the arising problem of the need of boundary conditions. In Section 5.4 we will see how to solve the given problem using the Finite Element Method. An implementation of the problem is given in Section 5.5 followed by

a numerical example on the unit circle in Section 5.6. Eventually, in Section 5.7 we introduce a modification of the method that shows a different possibility to formulate boundary conditions.

5.1 Preliminaries

Before we describe the workflow of the Ambient Signed Distance Method in detail we introduce some settings and analytical thoughts to obtain a better understanding of the situation. The main intention of this chapter is to give the reader an idea about the relation between intrinsic properties of a function defined on a submanifold and the corresponding properties of an extended function.

Let $G \subset \mathbb{R}^d$ be a bounded domain such that its boundary ∂G is a smooth compact submanifold of codimension one embedded in $\mathbb{R}^d$. Let $\omega := \partial G$ and let $\Gamma \subset \mathbb{R}^d$ be a tubular neighbourhood. See Section 4.1 for remarks on the tubular neighbourhood. The signed distance function gives the distance of a point in Γ to its closest point on ω.

Definition 9. *Let G, ω, and Γ be as above. The* signed distance function $\Phi\colon \Gamma \to \mathbb{R}$ *is given as*

$$\Phi(X) := \begin{cases} -\inf\limits_{x\in\omega} |x - X| & \text{if } X \in \Gamma \setminus G \\ \inf\limits_{x\in\omega} |x - X| & \text{if } X \in \Gamma \cap G\,. \end{cases}$$

The signed distance function is well-defined, continuous and differentiable inside any tubular neighbourhood. The following lemma gives two more useful properties of the signed distance function.

Lemma 5. *Let G, ω, and Γ be as above and let $\Phi\colon \Gamma \to \mathbb{R}$ be the signed distance function. Then Φ is differentiable on Γ and*

$$|\nabla\Phi| = 1.$$

Moreover, we have

$$\nabla^2\Phi \cdot \nabla\Phi = \boldsymbol{0}$$

where $\boldsymbol{0}$ is the zero-vector.

Proof. Let $\mathrm{cp}\colon \Gamma \to \omega$ be the closest point function as defined in Definition 5. Then

$$X = \mathrm{cp}(X) - \Phi(X)\,\nu(\mathrm{cp}(X))$$

for all $X \in \Gamma$. Taking the derivative we get

$$\mathbb{1} = D\,\mathrm{cp} - \nabla\Phi \cdot \nu^\top - \Phi\, D\,\mathrm{cp} \cdot D\nu\,,$$

where $D\,\mathrm{cp}$ and $D\nu$ are the Jacobi-matrices of cp and ν respectively. We multiply with $\nu^\top$ from the left and obtain with $\nu^\top \cdot D\,\mathrm{cp} = 0$

$$\nu^\top = \nu^\top \cdot D\,\mathrm{cp} - \nu^\top \nu\, \nabla\Phi^\top - \Phi\, \nu^\top \cdot D\,\mathrm{cp} \cdot D\nu = -\nabla\Phi^\top\,.$$

Therefore $|\nabla\Phi| = 1$.

Moreover we have for $i = 1, \ldots, d$

$$|\nabla\Phi| = 1 \quad \Rightarrow \quad \frac{\partial}{\partial x_i}\langle\nabla\Phi, \nabla\Phi\rangle = 0 \quad \Leftrightarrow \quad \langle\frac{\partial}{\partial x_i}\nabla\Phi, \nabla\Phi\rangle = 0 \quad \Leftrightarrow \quad \nabla^2\Phi \cdot \nabla\Phi = \mathbf{0}$$

□

An alternative proof of the first part of this theorem and other notes on the signed distance function can be found in [AD98].

Now, let $u \in L^2(\omega)$ be a function on the submanifold. Let E be the extension operator as described in Section 4.2, i.e., E extends functions on ω such that it is continuous in normal direction. Then for $U := Eu : \Gamma \to \mathbb{R}$ we have

$$U(X) = U(\mathrm{cp}(X)) = u(\mathrm{cp}(X)), \quad X \in \Gamma,$$

where cp is the closest point function (see Definition 5). We call U the *natural extension* of u.

The analysis of the presented method is based on the two following lemmata. They describe the relation between the intrinsic gradient of u and the gradient of U on the submanifold ω as well as the relation between the intrinsic divergence operator of a vector field on ω and the divergence of an extended vector field.

Lemma 6. *Let ω and Γ be as before. Let $u \in W_2^1(\omega)$ be a function on ω and $U : \Gamma \to \mathbb{R}$ be its natural extension. Then, for each point $x \in \omega$ the gradient of U lies in the tangential plane of ω and corresponds to the intrinsic gradient, i.e.,*

$$\nabla U(x) = \nabla_\omega u(x) \quad \textit{for all } x \in \omega.$$

Proof. Let $\nu(x)$ with $x \in \omega$ be the normal of ω at x. Since U is constant in normal direction, ∇U is tangential to ω. So $\nabla U(x) \cdot \nu(x) = 0$. Then we have for $x \in \omega$

$$\nabla_\omega u(x) = \nabla_p u(x) = \nabla U(x) - \nabla U(x) \cdot \nu(x)\nu(x) = \nabla U(x),$$

where ∇_p is the projected gradient (see Chapter 2). □

Lemma 7. *Let ω, Γ, and Φ be as before. Let $V : \Gamma \to \mathbb{R}^d$ be a vector field that is tangent at all level sets of the signed distance function Φ. Then, on ω the divergence of V coincides with the intrinsic divergence, i.e.,*

$$\nabla \cdot V(x) = \nabla_\omega \cdot V(x) \quad \textit{for all } x \in \omega.$$

This lemma turns out to be a special case of Lemma 11 of Section 6.1. Therefore, we skip the proof here but refer to the proof of Lemma 11 with $|\nabla \Phi| = 1$.

From these two properties we derive that the Laplace-Beltrami of $U|_\omega$ and the Laplacian of U coincide on ω. The methods of Ruuth and Merriman (see [RM08]) and Dzuik and Elliott (see [DE13]) are based on these two properties. They also work with the natural extension on some domain in a tubular neighbourhood.

The next step is to analyse how the gradient of U at some arbitrary point $X \in \Gamma$ can be related to the intrinsic gradient of u at $x = \mathrm{cp}(X)$. We define $H := \nabla^2 \Phi$ to be the Hessian matrix of Φ. Then, H gives the *curvature tensor* or also called the *embedded Weingarten map* of the level sets of Φ. See [Rei07] or [LR12] for a definition and notes on the curvature tensor. We note that the matrix H is symmetric.

Lemma 8. *Let ω, Γ, Φ, and H be as above. Let $h := H|_\omega$ be the restriction of H to the manifold , $X \in \Gamma$ be an arbitrary point and $x = \mathrm{cp}(X)$. Then, we have with $t := \Phi(X)$*

$$H(X) = h(x)\left(\mathbb{1} + t\, h(x)\right)^{-1}.$$

Proof. For $t = 0$, i.e., $X \in \omega$, the proposition is obviously true.

Let $\psi : \omega \times \mathbb{R} \to \Gamma$ be defined as

$$\psi(x,t) = x + t \cdot \nabla\Phi(x).$$

Then

$$X = \psi(x,t) \quad \text{for } x = \mathrm{cp}(X), t = \Phi(X).$$

We have to show that

$$\nabla^2\Phi(\psi(x,t)) = \nabla^2\Phi(x) \cdot (\mathbb{1} + t\nabla^2\Phi(x))^{-1}.$$

This is equivalent to

$$\nabla^2\Phi(\psi(x,t)) \cdot (\mathbb{1} + t\nabla^2\Phi(x)) = \nabla^2\Phi(x),$$

where the right hand side is independent of t. We consider one element and take the derivative with respect to t. Together with $\Phi(\psi(x,0)) = \Phi(x)$

$$\begin{aligned} & \sum_{j=1}^{d} \Phi_{ij}(\psi(x,t)) \cdot (\delta_{jk} + t\Phi(x)_{jk}) = \Phi(x)_{ik} \\ \Leftrightarrow \quad & \sum_{j=1}^{d} \nabla\Phi_{ij}(\psi(x,t))\psi_t(x,t) \cdot (\delta_{jk} + t\Phi_{jk}(x)) + \Phi_{ij}(\psi(x,t)) \cdot \Phi_{jk}(x) = 0 \end{aligned} \tag{5.1}$$

Now, it suffices to show (5.1) for one value t_0 of t. We set $t_0 = 0$. Then (5.1) simplifies to

$$\begin{aligned} & \sum_{j=1}^{d} \nabla\Phi_{ij}(x)\psi_t(x,t) \cdot \delta_{jk} + \Phi_{ij}(x) \cdot \Phi_{jk}(x) = 0 \\ \Leftrightarrow \quad & \sum_{\ell=1}^{d} \Phi_{ik\ell}(x)\Phi_\ell(x) + \sum_{j=1}^{d} \Phi_{ij}(x)\Phi_{jk}(x) = 0 \end{aligned}$$

If t_0 is not zero we can consider a shifted function $\tilde{\Phi} = \Phi - t_0$.

By Lemma 5 we have

$$\sum_{\ell=1}^{d} \partial_{i\ell}\Phi(x) \cdot \partial_\ell\Phi(x) = 0 \quad \text{for } i = 1, \ldots, d.$$

Differentiating this equation with respect to x_k we obtain

$$\sum_{\ell=1}^{d} \Phi_{ik\ell}(x)\Phi_\ell(x) + \Phi_{i\ell}(x)\Phi_{\ell k}(x) = 0$$

which is exactly what we needed to show. □

A geometrical proof of this theorem is given in [ZCMO96].

5.2 The Embedding PDE

In this chapter we concentrate on one important aspect of the method, namely how the embedding PDE is derived. In the previous chapter we already developed the basic idea.

Given a compact submanifold ω of codimension one and a linear elliptic second-order PDE on ω we search for a function $u \in W_2^2(\omega)$ that solves the given intrinsic PDE

$$L(x, u, \nabla_\omega u, \triangle_\omega u) = 0,$$

where ∇_ω and $\triangle_\omega$ denote the intrinsic gradient and the Laplace-Beltrami operator, respectively. First, we construct some ambient domain $\bar{\Omega}$. A detailed description of $\bar{\Omega}$ will be given in Section 5.3. For now, we take $\bar{\Omega}$ to be some domain, consisting of grid cells that lie within a tubular neighbourhood Γ of ω. We have to translate the given PDE into an embedding PDE. Therefore, we need to modify the gradient as well as the Laplace operator.

We now define a modified gradient and a modified divergence of functions defined on a tubular neighbourhood Γ.

Definition 10. *Let ω be as before and Γ a tubular neighbourhood. Let $Q \in \mathbb{R}^{d\times d}$ be a real square matrix. Now, let $U : \Gamma \to \mathbb{R}$ be a differentiable function and V a vector field on Γ. Then, we define the* modified gradient ∇_Q *as*

$$\nabla_Q U(X) := Q(X)\, \nabla U(X)\,.$$

Let ∇_Q^i denote the i-th component of ∇_Q. We define the modified divergence *as ∇_Q^i of the i-th entry of V,*

$$\nabla_Q \cdot V := \sum_{i=1}^{d} \nabla_Q^i V_i.$$

Before we present the next lemma we introduce a notation for the sum of the element-wise multiplication of two matrices of equal size:

Definition 11. *Let A, $B \in \mathbb{R}^{m\times n}$ be two real matrices of same size. We define*

$$A * B := \sum_{i,j} A_{ij} \cdot B_{ij}\,.$$

The following lemma gives another representation of the modified divergence.

Lemma 9. *Let Γ, Q, and V be as in Definition 10. Then*

$$\nabla_Q \cdot V = Q * J_V\,,$$

where J_V is the Jacobian matrix of V.

Proof. We have

$$\nabla_Q \cdot V = \sum_{i=1}^{d} \nabla_Q^i V_i = \sum_{i=1}^{d} (Q\, \nabla V_i)_i = \sum_{i=1}^{d} \left(\sum_{j=1}^{d} Q_{ij}\, (\nabla V_i)_j \right) = \sum_{i,j=1}^{d} Q_{ij}\, (J_V)_{ij}$$

which proves the lemma. □

The *modified Laplace operator* is the modified divergence of the modified gradient,

$$\triangle_Q U := \nabla_Q \cdot \nabla_Q U \,.$$

We have now defined a modified gradient and Laplace operator in general. Let us construct a matrix Q such that we can formulate the embedding PDE using this matrix. Let $Q \in \mathbb{R}^{d\times d}$ be defined as

$$Q(X) := (\mathbb{1} - \Phi(X)\, H(X))^{-1} \quad \text{for } X \in \Gamma,$$

where $H = \nabla^2\Phi$. The matrix Q describes the change of the gradient of $U(X)$ as X moves along the normal vector. Since Q equals the identity matrix on ω, we have

$$\nabla_Q U(x) = \nabla U(x) = \nabla_\omega u(x) \quad \text{for } x \in \omega.$$

Theorem 11. *Let ω, Γ, Φ, and H be as before. Let $u \in W_2^1(\omega)$ and $U\colon \Gamma \to \mathbb{R}$, $U = Eu$ be its natural extension. With $Q := (\mathbb{1} - \Phi\, H)^{-1}$ we have*

$$\nabla_\omega u(x) = \nabla_Q U(X) \quad \textit{for } x = \mathrm{cp}(X)\,.$$

Proof. Let X_0 be an arbitrary point in Γ and $x_0 = \mathrm{cp}(X_0)$. We consider a local, smooth, regular parametrisation $\chi : V \to W \cap \omega$ for an open connected set $V \subset \mathbb{R}^{d-1}$ and an open set $W \subset \mathbb{R}^d$ containing x_0. Let $\nu\colon \omega \to \mathbb{S}^{d-1}$ be the normal map and $\bar\nu : V \to \mathbb{S}^{d-1}$ be the Gauss map, that means $\nu(x) = -\nabla\Phi(x)$ and $\bar\nu(r) = \nu(\chi(r)) = -\nabla\Phi(\chi(r))$ for $r \in V$. We define

$$X(r,t) := \chi(r) - t\,\bar\nu(r) = \chi(r) + t\,\nabla\Phi(\chi(r)), \quad t \in \mathbb{R}, r \in V.$$

Let $J_\chi \in \mathbb{R}^{d\times(d-1)}$ be the Jacobian matrix of χ and $J_{\bar\nu} \in \mathbb{R}^{d\times(d-1)}$ be the Jacobian matrix of $\bar\nu$. Furthermore, let $h \in \mathbb{R}^{d\times d}$ be defined by $h = H|_\omega$, where H is the Hessian matrix of Φ. The following relationships hold:

$$J_{\bar\nu}(r) = -h(\chi(r)) \cdot J_\chi(r), \tag{5.2}$$
$$h(\chi(r)) = H(X(r,0)) \tag{5.3}$$
$$H(r,t) = h(\chi(r))\,(\mathbb{1} + t\,h(\chi(r)))^{-1}\,. \tag{5.4}$$

Let $\tilde J_X$ equal the first $d-1$ columns of the Jacobian matrix of X:

$$\tilde J_X := \begin{pmatrix} \frac{\partial X_1}{\partial r_1} & \cdots & \frac{\partial X_1}{\partial r_{d-1}} \\ \vdots & & \vdots \\ \frac{\partial X_d}{\partial r_1} & \cdots & \frac{\partial X_d}{\partial r_{d-1}} \end{pmatrix}.$$

Then $\tilde J_X = J_\chi - t\,J_{\bar\nu}$. Since U is constant in normal direction, we have for $r_0 = \chi^{-1}(x_0)$

$$U(X_0) = U(X(r_0,0)) = u(x_0)\,.$$

For the following calculation we omit the argument and introduce following convention: If two functions or vector spaces with different domains appear in one equation, we limit to the intersection of their domains. From this we obtain:

$$\nabla_\omega u^\top \cdot J_\chi = \nabla U^\top \cdot \tilde J_X = \nabla U^\top \cdot (J_\chi - tJ_{\bar\nu}) \overset{(5.2)}{=} \nabla U^\top \cdot (\mathbb{1} + t\,h) \cdot J_\chi\,.$$

With $\Phi(X(r,t)) = t$ we have

$$Q = (\mathbb{1} - \Phi H)^{-1} = (\mathbb{1} - t H)^{-1} \overset{(5.4)}{=} \left(\mathbb{1} - t\, h\, (\mathbb{1} + t\, h)^{-1}\right)^{-1}$$
$$= \left((\mathbb{1} + t\, h)(\mathbb{1} + t\, h)^{-1} - t\, h\, (\mathbb{1} + t\, h)^{-1}\right)^{-1} = \mathbb{1} + t\, h$$

and therefore

$$\nabla_\omega u^\top \cdot J_\chi = \nabla U^\top \cdot Q \cdot J_\chi \,.$$

Let us now consider the matrix

$$J_{\chi,\nu} := \begin{pmatrix} J_\chi, & \nu \end{pmatrix}.$$

This is a square matrix of full rank and therefore invertible. Moreover, we have

$$\nabla_\omega u(x_0)\, \nu(x_0) = 0 = U(X_0)\, \nu(x_0) = U(X_0)\, (\mathbb{1} + t\, h)\, \nu(x_0)\,.$$

So we have

$$\nabla_\omega u^\top \cdot J_{\chi,\nu} = \nabla U^\top \cdot Q \cdot J_{\chi,\nu} \,.$$

A multiplication from the right with $J_{\chi,\nu}^{-1}$ gives

$$\nabla_\omega u(x_0) = \nabla_Q U(X_0)\,.$$

□

Remark: This result can be found in [DE13] with a different proof.

Similar to the gradient, we find a relation between the Laplace-Beltrami operator of u and the Laplace operator of U.

Theorem 12. *Let ω, Γ, Φ, Q and H be as before. Let $u \in W_2^2(\omega)$ and $U\colon \Gamma \to \mathbb{R}$, $U = Eu$, be its natural extension. Then*

$$\triangle_\omega u(x) = \triangle_Q U(X) \quad \textit{for } x = \mathrm{cp}(X)\,.$$

Proof. We consider the same setting as in the proof of the previous lemma. We have already shown that $\nabla_Q U(X(r,t)) = \nabla_\omega u(\chi(r))$. This leads to

$$(D\nabla_\omega u)(\chi(r)) \cdot J_\chi(r) = (D\nabla_Q U)(X(r,t)) \cdot \tilde{J}_X(r,t)$$

for all $t \in [-\varepsilon, \varepsilon]$. Moreover, we have $\tilde{J}_X(r,t) = Q(r,t) \cdot J_\chi(r)$ and therefore

$$(D\nabla_\omega u)(\chi(r_0)) \cdot J_\chi = (D\nabla_Q U)(X(r_0,t)) \cdot Q \cdot J_\chi.$$

The matrix J_χ is of full rank and so, considered as a map, $J_\chi : \mathbb{R}^{d-1} \to \mathbb{R}^d$ is injective. This implies

$$(D\nabla_\omega u)(\chi(r_0)) = (D\nabla_Q U)(X(r_0,t)) \cdot Q.$$

Since Q is invertible we have

$$(D\nabla_\omega u)(x_0) \cdot Q^{-1}(X_0) = (D\nabla_Q U)(X_0). \tag{5.5}$$

Then

$$\triangle_Q U = Q * (D\, \nabla_Q U) \overset{(5.5)}{=} \sum_{i,j} Q_{ij} \cdot (D\nabla_\omega u\, Q^{-1})_{ij} = \sum_{i,j,k} Q_{ij} \cdot Q^{-1}_{kj} \cdot (D\nabla_\omega u)_{ik}$$
$$\overset{(\#)}{=} \sum_{i,k} \delta_{ik} \cdot (D\nabla_\omega u)_{ik} = \mathrm{trace}(D\nabla_\omega u) = \triangle_\omega u\,,$$

where we used the symmetry of Q at $(\#)$. □

The two previous theorems enable us to translate the given problem on the submanifold into a problem in some domain $\bar{\Omega}$ contained in a tubular neighbourhood Γ. The intention is to formulate a PDE on $\bar{\Omega}$ such that its solution U equals u on ω and at the same time is constant in the normal direction ν.

Theorem 13. *Let ω, $\bar{\Omega}$, and Q be as before. Let L be a linear elliptic differential operator. If $U : \bar{\Omega} \to \mathbb{R}$ is constant in normal direction and solves the linear elliptic PDE*

$$L(\mathrm{cp}(X), U(X), \nabla_Q U(X), \triangle_Q U(X)) = 0$$

on $\bar{\Omega}$ then its restriction $u = U|_\omega$ solves the intrinsic PDE

$$L(x, u(x), \nabla_\omega u(x), \triangle_\omega u(x)) = 0.$$

Proof. Let U be constant in normal direction and $u = U|_\omega$. By the lemmata proven above we have for $x = (\mathrm{cp}(X))$ and $\alpha_{ij}, \beta_i \in \mathbb{R}$ for $i, j = 1, \ldots, d$

$$\begin{aligned}\alpha_{ij}(\mathrm{cp}(X))\triangle_Q U(X)_{ij} &= \alpha_{ij}(x)\triangle_\omega u(x)_{ij}\\ \beta_i(\mathrm{cp}(X))\nabla_Q U(X)_i &= \beta_i(x)\nabla_\omega u(x)_i\,.\end{aligned}$$

Therefore, we have

$$\begin{aligned}L(x, u, \nabla_\omega u, \triangle_\omega u) &= \sum_{i,j=1}^{d} \alpha_{ij}(x)\triangle_\omega u(x)_{ij} + \sum_{i=1}^{d} \beta_i(x)\nabla_\omega u(x)_i + \gamma u(x)\\ &= \sum_{i,j=1}^{d} \alpha_{ij}(\mathrm{cp}(X))\triangle_Q U(X)_{ij} + \sum_{i=1}^{d} \beta_i(\mathrm{cp}(X))\nabla_Q U(X)_i + \gamma U(X) = L(\mathrm{cp}(X), U, \nabla_Q U, \triangle_Q U)\end{aligned}$$

□

We note that by this construction of the embedding PDE ellipticity is preserved. In the method presented in [DDEH10] and [DE13] elliptic regularity is lost because they involve a projection. The projection matrix $P = \mathbb{1} - \nu \otimes \nu$ has a zero eigenvector

$$P\nu = 0.$$

That means there is no diffusion in the normal direction of the submanifold and ellipticity is restricted to each level set of Φ. The Ambient Signed Distance Method presented here avoids any kind of projection. The matrix Q is positive definite. We have

$$Q = (\mathbb{1} - \Phi H)^{-1} = \mathbb{1} + t\,h \quad \text{with } t \in [-\varepsilon, \varepsilon].$$

The eigenvalues of Q are

$$1 + t\kappa$$

where κ are the main curvatures of ω. Since $\varepsilon < \inf_{x\in\omega} \frac{1}{|\kappa|_{\max}(x)}$ where $|\kappa|_{\max}$ is the absolute value of the biggest normal curvature we can follow that

$$1 + t\kappa > 0.$$

Let us now have a look at an example.

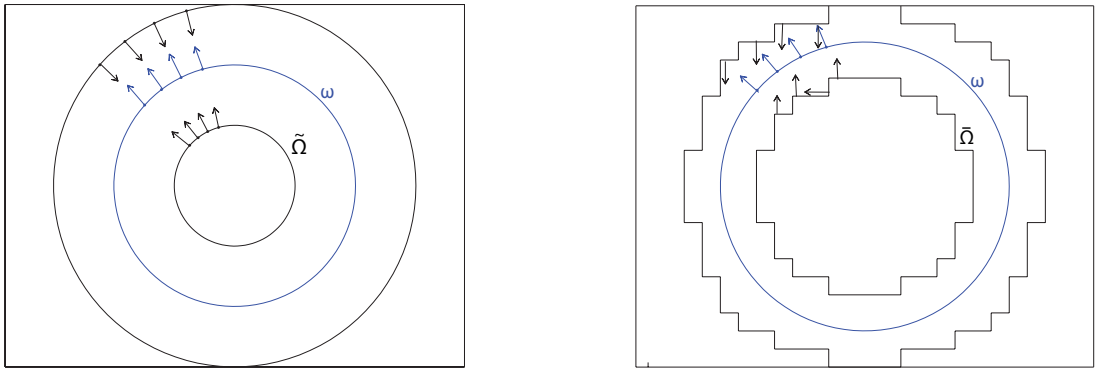

Figure 5.1: *left:* normal directions on $\partial\tilde{\Omega}$ coincide with normal directions on ω, *right:* for $\bar{\Omega}$ the normal directions do not coincide

Example 1. *Let ω be a given submanifold embedded in $\mathbb{R}^d$. Consider the following linear elliptic PDE on ω:*

$$-\triangle_\omega u + c\,u = f\ ,$$

for a fixed constant $c > 0$ and a function $f \in L_2(\omega)$. Let E be the extension operator as before. Then the embedding PDE on $\bar{\Omega}$ equals

$$-\triangle_Q U + c\,U = Ef$$

with $Q = (\mathbb{1} - \Phi H)^{-1}$.

The embedding PDE is defined on a domain in $\mathbb{R}^d$. For the well-posedness of the embedding PDE we need to add boundary conditions. Let

$$\tilde{\Omega} := \{x + t\nu(x) \,|\, x \in \omega,\ t \in [\delta_1, \delta_2]\}, \qquad \delta_1 < 0 < \delta_2\,,$$

where ν is the normal on ω. Let δ_1 and δ_2 be small enough such that $\tilde{\Omega}$ is included in a tubular neighbourhood of ω. Then the well known Neumann boundary conditions

$$\frac{\partial U(x)}{\partial \nu(x)} = 0, \quad x \in \partial\Omega,\ \nu(x) \text{ being the normal direction on } \partial\Omega,$$

formulate the condition that U is constant along the normals on $\partial\tilde{\Omega}$. Then the embedding PDE together with the Neumann boundary condition form a well posed problem.

As we had mentioned before, the domain $\bar{\Omega}$ is a union of grid cells. Therefore, it is not of the same structure as $\tilde{\Omega}$. Consequently, the Neumann boundary conditions cannot be used in the standard way but have to be modified. In the next chapter we present a formulation of boundary conditions for domains of arbitrary structure.

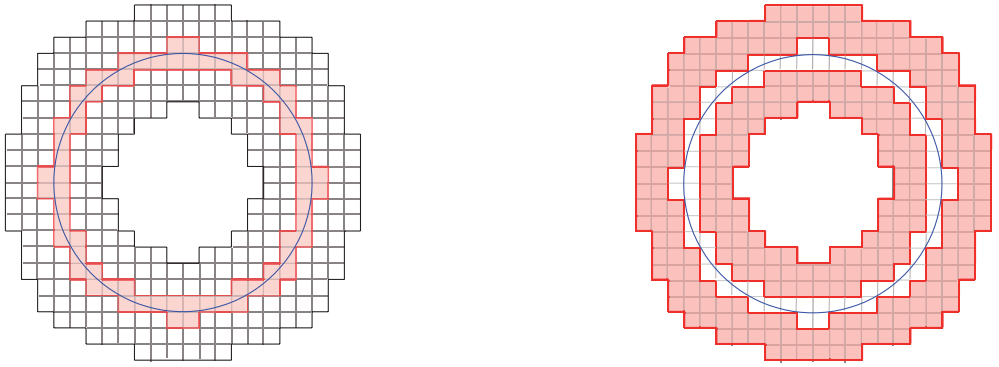

Figure 5.2: *left:* active cells, *right:* outer part of $\bar{\Omega}$

5.3 Boundary Conditions

Since we are now solving a PDE on a domain in $\mathbb{R}^d$ we need to define boundary conditions. As just mentioned the boundary conditions cannot be formulated in a standard way. The domain $\bar{\Omega}$ in the Ambient Signed Distance Method is constructed as the union of grid cells. Therefore, the normals on $\partial\bar{\Omega}$ do not coincide with the normal direction on ω (see Figure 5.1). We need to find a new version of boundary conditions. In this thesis we present two different ways to handle boundary conditions. In this section we present a new formulation of boundary conditions which can be illustrated as a "thickening" of the boundary. The alternative way follows in Section 5.7.

In the Ambient B-spline Method (Chapter 4) the ambient domain Ω contained all active cells (see Definition 4). Now, we extend this domain by cells, such that they represent a d-dimensional boundary of the domain.

Definition 12. *Let ω be as before and G be a bi-infinite uniform grid. Let $\sigma_{act} \subset G$ denote all active cells according to Definition 4 and I be the index set of all active B-splines, that is*

$$I := \{i \in \mathbb{Z}^d \,|\, \text{supp}(b_i) \cap \sigma_{act} \neq \emptyset\}.$$

The extended ambient domain $\bar{\Omega}$ *is defined as*

$$\bar{\Omega} := \{\sigma \in G \mid \sigma \subset \text{supp}(b_i) \text{ for } i \in I\}.$$

We have two conditions on U on the extended ambient domain $\bar{\Omega}$. First, U should fulfil the embedding PDE as formulated in the previous chapter and second, U should be constant in normal direction. We formulate the second condition as

$$\nabla U^\top \cdot \nabla \Phi = 0\,.$$

The basic idea is to divide the grid cells σ in $\bar{\Omega}$ into two groups: On cells near $\partial\bar{\Omega}$ we claim $\nabla U^\top \cdot \nabla \Phi = 0$, while on the cells near ω we want U to solve the embedding PDE. Finally, both equations are solved using the Finite Element Method.

Therefore, $\bar{\Omega}$ is divided into an *inner part* and an *outer part*. The active cells make the inner part and the outer part consists of all remaining cells in $\bar{\Omega}$. Figure 5.2 shows an example for ω being the unit circle embedded in $\mathbb{R}^2$ and B-spline order $n = 3$.

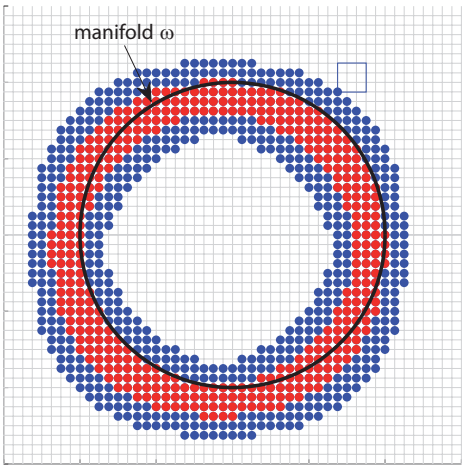

Figure 5.3: relevant B-splines are divided into two groups

Now, we find all B-splines $b_k, k \in K$ with $\text{supp}(b_K) \cap \bar{\Omega} \neq \emptyset$ for an index set K. These B-splines are called *relevant B-splines*. The relevant B-splines of order n span the space $S^n(\bar{\Omega})$. Finally, we define $J := K \setminus I$, i.e., all relevant B-splines that are not active B-splines.

See Figure 5.3 for a better understanding of the current situation. In this example the submanifold ω is the unit circle again and the order n of the B-splines equals three. Each B-spline is indicated by its lower left corner. All active B-splines b_i, $i \in I$, are marked with red dots and the B-splines b_j, $j \in J$ are marked blue. We marked the support of one B-spline with a blue rectangle.

We will now use this setting to solve the two conditions on $\bar{\Omega}$ according to the following apportionment:

- on the inner part of $\bar{\Omega}$ we solve the embedding PDE
- on the outer part we solve $\nabla U^\top \cdot \nabla \Phi = 0$.

To solve this we use the Finite Element Method as introduced in Section 2.2.

5.4 Finite Element Method

We have divided the extended ambient domain $\bar{\Omega}$ into two disjoint sets. On the inner part we want to solve the embedding PDE, whereas in the outer part we require U to solve $\nabla U^\top \cdot \nabla \Phi = 0$. We now use the Finite Element Method to solve these equations. We restrict the solution space to the spline space $S^n(\bar{\Omega})$. This spline space is the span of all relevant tensor product B-splines of order n.

Before doing so, we construct the weak formulation of the equations we have to solve. Let us again consider the model PDE for a better understanding: Let ω be the unit circle embedded in $\mathbb{R}^2$ and $u\colon \omega \to \mathbb{R}$ be the solution of

$$-\triangle_\omega u + c\, u = f$$

for some continuous function $f : \omega \to \mathbb{R}$ and a constant $c > 0$. The embedding PDE has the form

$$-\triangle_Q U + c\, U = F,$$

where $F := Ef$. Multiplying by a test function and integrating over $\bar{\Omega}$ gives the weak formulation. We want U to solve

$$-\int_{\bar{\Omega}} \triangle_Q U \cdot g + c \int_{\bar{\Omega}} U \cdot g = \int_{\bar{\Omega}} F \cdot g, \quad \text{for all } g \in L_2(\bar{\Omega}).$$

At the same time we formulate the weak formulation of the condition $\nabla U^\top \cdot \nabla \Phi = 0$ as

$$\int_{\bar{\Omega}} (\nabla U^\top \cdot \nabla \Phi)\, g = 0 \quad \text{for all test functions } g \in L_2(\bar{\Omega}).$$

Now, we limit our solution space to the spline space $S^n(\bar{\Omega})$ based on the grid G. Then, the test functions g are elements of the spline space $S^n(\bar{\Omega})$ and U is approximated by

$$U_h = \sum_{k \in K} \xi_k b_k^n \in S^n(\bar{\Omega}) \tag{5.6}$$

with K being the index set of all relevant B-splines and ξ_k the coefficients of the spline.

The equations change to

$$-\int_{\bar{\Omega}} \triangle_Q U_h \cdot g + c \int_{\bar{\Omega}} U_h \cdot g = \int_{\bar{\Omega}} F \cdot g \tag{5.7}$$

and

$$\int_{\bar{\Omega}} (\nabla U_h^\top \cdot \nabla \Phi)\, g = 0\,, \quad \text{for all } g \in S^n(\bar{\Omega}). \tag{5.8}$$

Since the relevant tensor product B-splines form a finite basis of the spline space $S^n(\bar{\Omega})$ it suffices to use them as test functions. We want (5.7) to hold on the inner part and (5.8) on the outer part. Therefore we test (5.7) with the active B-splines b_i, $i \in I$ and use the B-splines b_j, $j \in J$ to test (5.8).

Then, with (5.6), we obtain the following set of equations:

$$\sum_k \xi_k \left(-\int_{\bar{\Omega}} \left(\triangle_Q b_k\right)\, b_i + c \int_{\bar{\Omega}} b_k\, b_i \right) = \int_{\bar{\Omega}} F b_i \qquad \text{for } i \in I$$

$$\sum_k \xi_k \int_{\bar{\Omega}} \left(\nabla b_k^\top \cdot \nabla \Phi\right) b_j = 0 \qquad \text{for } j \in J.$$

Since $I \dot{\cup} J = K$, the number of variables and equations are identical. The term $\int_\Omega \triangle_Q b_k\, b_i$ is unhandy because we need to estimate the value of $D(Q \cdot \nabla b_k)$. Moreover, this term eliminates the linear case $n = 2$ by using the second derivative of the B-splines. Therefore, we reformulate the term and substitute $\triangle_Q U_h$ using the divergence theorem.

Let v be a twice differentiable function on Ω. Let $q^{ij} := Q_{ij}$ and lower indices denote derivatives, that means $q^{ij}_k := \frac{\partial Q_{ij}}{\partial x_k}$. Then

$$\begin{aligned}
\triangle_Q v = \nabla_Q \cdot \nabla_Q v = \sum_{i,j} q^{ij} \left(\sum_k q^{ik} v_k \right)_j = \sum_{i,j,k} q^{ji} q^{ik}_j v_k + \sum_{i,j,k} q^{ji} q^{ik} v_{kj} \\
= \sum_{i,j,k} \left(q^{ji}_j q^{ik} + q^{ji} q^{ik}_j \right) v_k + \sum_{i,j,k} q^{ji} q^{ik} v_{kj} - \sum_{i,j,k} q^{ji}_j q^{ik} v_k = \sum_{i,j,k} \left(q^{ji} q^{ik} v_k \right)_j - \sum_{i,j,k} q^{ji}_j q^{ik} v_k \\
= \sum_j \left(\sum_k \left(Q^2 \right)^{jk} v_k \right)_j - \sum_{i,j,k} q^{ji}_j q^{ik} v_k = \operatorname{div} \left(Q^2 \nabla v \right) - \sum_{i,j,k} q^{ji}_j q^{ik} v_k
\end{aligned}$$

Here we used the symmetry of Q.

Let R be a vector with $R_k = \sum_{i,j} q^{ij}_j q^{ik}$. Then we have

$$\triangle_Q v = \operatorname{div} \left(Q^2 \nabla v \right) - R \nabla v \, .$$

So, we can substitute the first integral to obtain:

$$\sum_k \xi_k \left(- \int_{\bar{\Omega}} \operatorname{div} \left(Q^2 \nabla b_k \right) \cdot b_i + \int_{\bar{\Omega}} R \nabla b_k \cdot b_i + c \int_{\bar{\Omega}} b_k \cdot b_i \right) = \int_{\bar{\Omega}} F \cdot b_i \, .$$

Now, we use the divergence theorem, which is a generalisation of Greens formula, to obtain

$$\sum_k \xi_k \left(\int_{\bar{\Omega}} \nabla_Q b_k \cdot \nabla_Q b_i - \int_{\partial \bar{\Omega}} \left(Q^2 \nabla b_k \cdot \nu \right) b_i + \int_{\bar{\Omega}} R \, \nabla b_k \cdot b_i + c \int_{\bar{\Omega}} b_k \cdot b_i \right) = \int_{\bar{\Omega}} F \cdot b_i,$$

where ν is the normal.

The second term integrates over the boundary of $\bar{\Omega}$. Since every active B-spline is zero at the boundary of $\bar{\Omega}$ this term vanishes. We finally get:

$$\begin{aligned}
&\sum_k \xi_k \left(\int_{\bar{\Omega}} \nabla b_k^\top \cdot Q^2 \cdot \nabla b_i + \int_{\bar{\Omega}} R \cdot \nabla b_k \, b_i + c \int_{\bar{\Omega}} b_k \, b_i \right) = \int_{\bar{\Omega}} F \, b_i && \text{for } i \in I \\
&\sum_k \xi_k \int_{\bar{\Omega}} \left(\nabla b_k \cdot \nabla \Phi \right) \, b_j = 0 && \text{for } j \in J
\end{aligned}$$

We combine these equations and write in matrix form

$$\left(S + Z + cM + T \right) \xi = r$$

with stiffness matrix

$$S_{ij} = \begin{cases} \int_{\bar{\Omega}} \nabla_Q b_i \cdot \nabla_Q b_j & \text{if } i \in I \\ 0 & \text{else,} \end{cases}$$

mass matrix

$$M_{ij} = \begin{cases} \int\limits_{\bar{\Omega}} b_i b_j & \text{if } i \in I \\ 0 & \text{else,} \end{cases}$$

extra term

$$Z_{ij} = \begin{cases} \int\limits_{\bar{\Omega}} \left(R \cdot \nabla b_j\right) b_i & \text{if } i \in I \\ 0 & \text{else,} \end{cases}$$

outer term matrix

$$T_{ij} = \begin{cases} 0 & \text{if } i \in I \\ \int\limits_{\bar{\Omega}} \left(\nabla b_j \cdot \nabla \Phi\right) b_i & \text{else,} \end{cases}$$

and the right hand side

$$r_i = \begin{cases} \int\limits_{\bar{\Omega}} F b_i & \text{if } i \in I \\ 0 & \text{else.} \end{cases}$$

We can now solve this linear system of equations to obtain the control points ξ. We get a spline $U_h \in S^n(\bar{\Omega})$. Before we have a look at the implementation and numerical results we share some thoughts on the theoretical analysis of the Ambient Signed Distance Method:

Corollary 3. *Let ω and $\bar{\Omega}$ be as before. If the given intrinsic PDE is solvable then the embedding PDE is solvable. If the intrinsic PDE be uniquely solvable and $U : \bar{\Omega} \to \mathbb{R}$ and $\tilde{U} : \bar{\Omega} \to \mathbb{R}$ are constant in normal direction and solve the embedding PDE then $U = \tilde{U}$ on $\bar{\Omega}$.*

Proof. Let u be the solution of the given PDE. Then, the natural extension Eu is a solution of the embedding PDE.

Let U and $\tilde{U}$ be constant in normal direction. Then $U - \tilde{U}$ is constant in normal direction. Moreover, for all $x \in \omega$

$$(U - \tilde{U})(x) = U(x) - \tilde{U}(x) = u(x) - u(x) = 0.$$

Therefore, $(U - \tilde{U})(X) = 0$ for all $X \in \bar{\Omega}$. □

5.5 Implementation

In this chapter we describe how we implemented the presented method using MATLAB. Here, we concentrate on the PDE

$$-\triangle_\omega u + u = f \ .$$

The program is implemented in such a way that it receives as INPUT

- a smooth compact manifold ω to which the signed distance function is given,

- a grid width $h > 0$,
- an order $n \in \mathbb{N}$ of the B-splines and
- a continuous function $f : \omega \to \mathbb{R}$.

The OUTPUT of the program is

- the index set K of the relevant B-splines,
- a vector $(\xi_k)_{k \in K}$ containing the control points of the resulting spline U and
- the arising error on the manifold ω.

The returned error consists of the maximal error as well as the root mean square error.

First of all, we construct a uniform grid with grid width $h > 0$ and compute the extended ambient domain $\bar{\Omega}$ as well as the active and relevant B-splines. The active B-splines are computed in the same way as it was done in the Ambient B-spline Method (see Section 4.3). Let I be the index set of all active B-splines. To obtain the index set K of all relevant B-splines we add all $\nu \in \mathbb{N}^d$ with $\nu_j = 1, \dots, (n-1)$ to each $i \in I$. With the call *unique* in MATLAB repeated indices are deleted. Then we compute the entries of the matrices. This is the main part of the program and requires about 90% of the calculation time. All matrices are sparse but of big size. Therefore, we construct them using the "sparse-method" that MATLAB provides.

To compute the mass matrix M with

$$M_{ij} = \int_{\bar{\Omega}} b_i \cdot b_j, \quad i \in I, j \in K$$

for all active B-splines b_i and relevant B-splines b_j we make use of two properties: First, the entries in the matrix repeat, and second, the integration can be done in each coordinate direction separately. The first property results from the fact that in case of a uniform grid the B-splines are translations of each other, i.e.,

$$b_i(t) \cdot b_j(t) = b_{i+\ell}(t + h\ell) \cdot b_{j+\ell}(t + h\ell)$$

with $i, j, \ell \in \mathbb{Z}^d$ and $i + \ell := [i_1 + \ell_1, \dots, i_d + \ell_d]$. So, we precompute the values first and then insert them in the matrix at the correct positions.

To compute the values we make use of the second property. First of all, we notice that due to the local support of the B-splines the integral needs to be taken only over the common support of the affected B-splines. Let $\bar{\Omega}_{ij} := \operatorname{supp}(b_i) \cap \operatorname{supp}(b_j)$. This is a d-dimensional interval where each side of $\bar{\Omega}_{ij} = \bar{\Omega}_{ij}^1 \times \ldots \times \bar{\Omega}_{ij}^d$ has a maximal length of $n \cdot h$. Then

$$M_{ij} = \int_{\bar{\Omega}_{ij}} b_i \cdot b_j \,.$$

Moreover, each d-dimensional tensor product B-spline b_k is the product of d one dimensional B-splines (see Section 3.2). Therefore, we have

$$\int_{\bar{\Omega}_{ij}} b_i(x_1, \dots, x_d) \cdot b_j(x_1, \dots, x_d) = \prod_{\ell=1}^{d} \int_{\bar{\Omega}_{ij}^\ell} b_i(x_\ell) \cdot b_j(x_\ell).$$

That means, we can integrate in each direction separately. To compute the integral we use Gauss quadrature. This is a numerical method to compute integrals by a weighted sum of function values:

$$\int_a^b f(x)dx \approx \frac{b-a}{2} \sum_{i=1}^{m} w_i f\left(\frac{b-a}{2} z_i + \frac{b+a}{2}\right)$$

with order $m \in \mathbb{N}$, precomputed weights w_i and points $z_i \in [-1,1]$. We choose $m = 4$ so that Gauss Quadrature is accurate for polynomials up to order 8. In this way the mass matrix can be computed in less calculation time.

The term in the stiffness matrix

$$S_{ij} = \int_{\bar{\Omega}} \nabla_Q b_i \cdot \nabla_Q b_j$$

and the extra term

$$Z_{ij} = \int_{\bar{\Omega}} \left(R \cdot \nabla b_j\right) b_i$$

are united in one matrix. Again, we restrict the integration domain to $\bar{\Omega}_{ij}$ and save the matrix in sparse form. We get

$$(S+Z)_{ij} = \int_{\bar{\Omega} ij} \nabla b_i^\top \cdot Q^2 \cdot \nabla b_j + \left(R \cdot \nabla b_j\right) b_i \,.$$

This integral is computed numerically using Romberg's method. This is an iterative method that works like a repetition of the trapezium rule or the rectangle rule. We used a multidimensional version of Romberg's method as presented in [EB93].

Even the values of the outer term

$$T_{ij} = \int_{\bar{\Omega}} \left(\nabla b_j^\top \nabla \Phi\right) b_i, \quad i \in J, j \in K$$

are computed using Romberg's method. Here, the integration area is not necessarily a rectangle because $\bar{\Omega}$ does not necessarily contain the whole support of the affected B-splines. So we have to integrate over

$$\bar{\Omega}_{ij} \cap \bar{\Omega} = \mathrm{supp}(b_i) \cap \mathrm{supp}(b_j) \cap \bar{\Omega} \,.$$

This integration area is a union of grid cells. Therefore, we integrate over each cell and sum up the result.

After computing the matrices $M, (S+Z)$, and T and the vector r containing the right hand side, we can use the backslash operator to solve the system of linear equations $(M+S+Z+T)\,\xi = r$ for the control points $\xi = (\xi_k)_{k \in K}$.

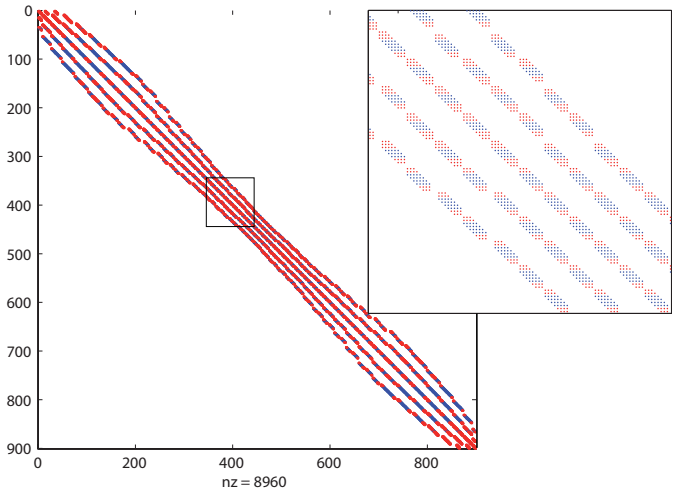

Figure 5.4: the matrix $M + S + Z + \tilde{T}$ for $n = 3$ and $h = \frac{1}{16}$

Finally, the spline $U = \sum_{k \in K} \xi_k b_k$ is evaluated at a finite number of discrete points p_i on the manifold. If the exact solution is known the error can be computed by comparing the values of the spline to the original solution u at these points. We compute the maximal error

$$\max_i \left| \sum_k \xi_k b_k(p_i) - u(p_i) \right|, \quad p_i \in \omega,$$

as well as the root mean square error

$$\sqrt{\frac{\sum_i \left(\sum_k \xi_k b_k(p_i) - u(p_i) \right)^2}{\# p_i}}, \quad p_i \in \omega,$$

where $\# p_i$ is the number of data points p_i. If the solution u is not given, one can compute the error by testing how good the resulting spline solves the given PDE. Therefore, the spline is inserted in the given intrinsic PDE. Then, the optimal order of convergence that we can expect is $\mathcal{O}(h^{n-2})$ for second-order PDEs and $\mathcal{O}(h^{n-1})$ for first order PDEs. In the examples presented in the following chapter the exact solution is known. Therefore, the error is computed by comparing the spline with the exact solution.

5.6 Numerical Results

We applied the Ambient Signed Distance Method to solve intrinsic PDEs on manifolds on the following 2-D example. We used an implementation in MATLAB as described in the previous chapter.

The manifold ω is the unit circle embedded in $\mathbb{R}^2$. The signed distance function is given as

$$\Phi(x, y) = \sqrt{x^2 + y^2} - 1 \,.$$

The Hessian matrix equals

$$H = \frac{1}{(x^2 + y^2)^{\frac{3}{2}}} \begin{pmatrix} y^2 & -xy \\ -xy & x^2 \end{pmatrix}$$

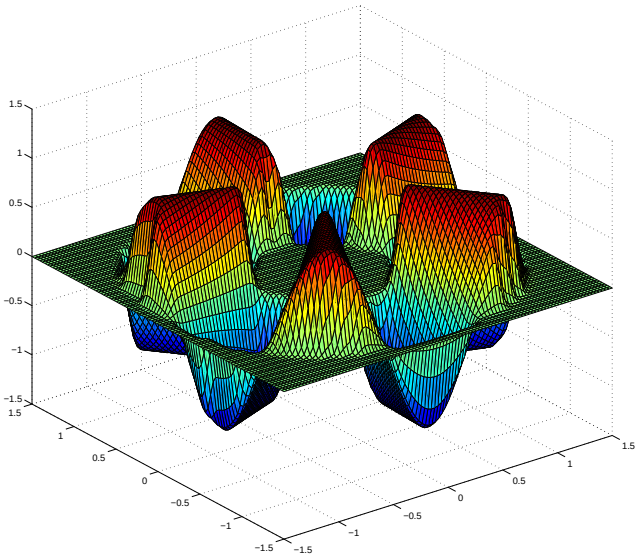

Figure 5.5: 3-D plot of the result on the extended ambient domain

and the matrix Q is

$$Q = \frac{1}{x^2+y^2}\begin{pmatrix} \sqrt{x^2+y^2}\,y^2+x^2 & -\left(\sqrt{x^2+y^2}-1\right)xy \\ -\left(\sqrt{x^2+y^2}-1\right)xy & \sqrt{x^2+y^2}\,x^2+y^2 \end{pmatrix}.$$

The matrix Q has to be computed once in advance via, for example, MAPLE. We consider the model PDE with $c = 1$ and $f = 26\,\cos(5\,\theta)$ for θ being the angle. So, the given intrinsic PDE is

$$-\triangle_\omega u + u = 26\,\cos(5\,\theta)$$

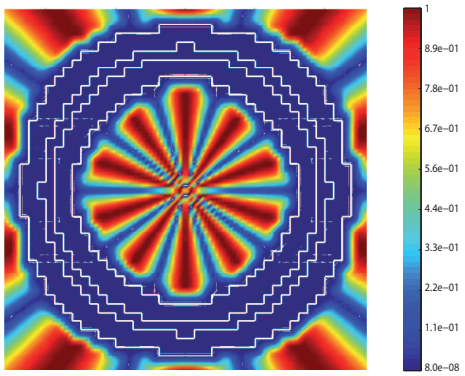

Figure 5.6: the error

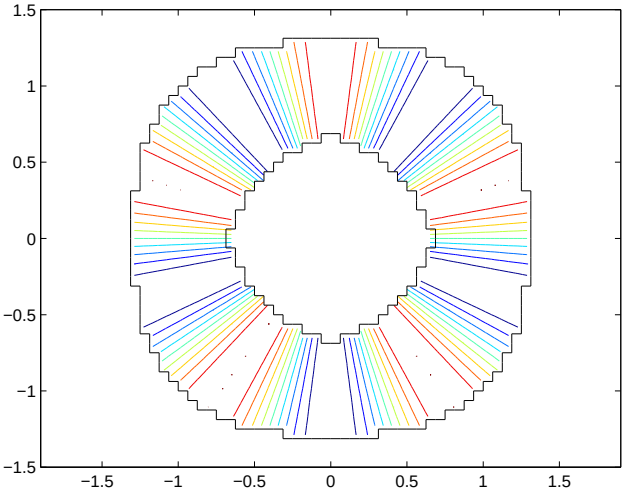

Figure 5.7: contour lines

on ω. The embedding PDE is of the form

$$-\triangle_Q U + U = 26\cos(5\theta)$$

defined on $\bar{\Omega}$. We applied our method for the orders $n = 3$, 4, and 5 for grid widths $h = \frac{1}{2^m}$, $m = 4, 5, 6, 7$. For bigger grid widths the extended ambient domain exceeds the tubular neighbourhood.

The values of the matrices converge differently concerning the grid width h. For 2-D problems the magnitude of the values of the stiffness matrix S_{ij} are independent of h. The mass matrix M converges in $\mathcal{O}(h^2)$ and the extra term Z in $\mathcal{O}(h)$. Therefore, the sum of these matrices $(M+S+Z)$ does not converge with h. The values of the outer term T_{ij} converge in $\mathcal{O}(h)$. For very small values of h the condition of the matrix $(M+S+Z+T)$ becomes very high. For $n = 5$ and $h = \frac{1}{64}$ the matrix condition is in the magnitude of 10^{10}. A very high matrix condition can influence the solution negatively. Therefore, the outer term T is normalised by the following rule: Each row of the outer term T is normalised separately in such a way that the entries on the diagonal equal 1. For each row $i \in J$, we set

$$\tilde{T}_{ij} = \frac{T_{ij}}{T_{ii}}.$$

After doing so the matrix condition for $n = 5$ and $h = \frac{1}{64}$ is in the magnitude of 10^6.

In Figure 5.4 we see the sparsity pattern of the matrix $(S + Z + M + \tilde{T})$ for $n = 3$ and $h = \frac{1}{16}$. The matrix is a sparsely filled square matrix. The blue dots indicate positions where the stiffness matrix, mass matrix and the extra term are non-zero whereas the red dots show where the outer term is non-zero. By construction each row contains only one colour but each column is of mixed colour. The resulting pattern consists of 5 ribbons of maximal widths of 5 points. This is because each B-spline of order 3 has 5×5 B-splines with which it has a non vanishing common support.

The resulting spline U_h is shown in Figure 5.5 as a 3-D plot. We used the build-in method *surf* of MATLAB to obtain this plot. We can see that in a band around the unit circle the spline takes values between -1 and 1. Further away the function is constantly 0.

The exact solution u of the intrinsic PDE equals $u = \cos(5\,\theta)$. Figure 5.6 depicts the maximal error between the exact solution of the embedding PDE and the approximation obtained by using the Ambient Signed Distance Method. The inner and outer part of $\bar{\Omega}$ are marked with a white line.

In Figure 5.7 we can see the contour lines of the spline U_h for grid width $h = \frac{1}{16}$ and order $n = 4$. The contour lines look like straight lines. This means that ∇U_h is perpendicular to the circle, which was intended. The technique of the modified boundary conditions does the trick. To compute the error we chose about 400 equally spread points p_i, $i = 1, \ldots, 400$, on the unit circle. On these points the maximal and the root mean square error were computed as described in the previous chapter.

A theoretical proof of the error convergence is still open. Yet, we share some thoughts here. In the error analysis of the Ambient B-spline Method we had proven an error convergence in $\mathcal{O}(h^n)$. The optimal error behaviour in the extended domain remains even after limiting to the submanifold again. Moreover, we know by the Céa-Lemma that the standard Finite Element Method returns an approximation that is as good as the best approximation in S^n for $p = 2$. It should be possible to transfer the main ideas of the proof in Section 4.2 and a suitable version of Céa-Lemma to formulate a proof on the convergence order of the Ambient Signed Distance Method.

Figure 5.8 illustrates the error in respect to the grid width. The red line depicts the maximal error and the blue line indicates the root mean square error. We can see that the method gives good results. For $h \to 0$ we have $U_h|_\omega \to u$. The error converges in $\mathcal{O}(h^n)$ for $n = 3$ and 5. Most of the existing approximation methods work with linear approximation and therefore have a convergence in $\mathcal{O}(h^2)$. An error in the magnitude of 10^{-10} as we have for $n = 5$ and $h = \frac{1}{128}$ is hardly reachable for presently existing methods.

5.7 Modification

The Ambient Signed Distance Method as introduced in this thesis is easily implemented and provides good results. Yet, for a high order n the band, i.e., the extended ambient domain, around the manifold becomes very thick. It might even be possible that the extended ambient domain is no longer contained in any tubular neighbourhood of the manifold. In that case we can choose a smaller grid width h to reduce the band width. This results in a very fine grid, maybe even finer than needed. To avoid these problems we introduce a modification of the method in this section. Unfortunately, this modification shows some difficulties.

The setting of the method remains the same. A difference appears in the discretisation of the problem. We do not divide the extended ambient domain $\bar{\Omega}$ into two parts and consider the embedding PDE and the boundary condition separately. Instead, we claim U to fulfil both equations on the domain.

Let G be a uniform grid of grid width h and Ω is the union of all active cells as defined in Definition 4. The finite element space is the spline space $S^n(\Omega)$ spanned by the active B-splines. Again concentrating on the model PDE we solve

$$-\triangle_Q U + cU = F$$

as well as

$$\nabla U^\top \cdot \nabla \Phi = 0$$

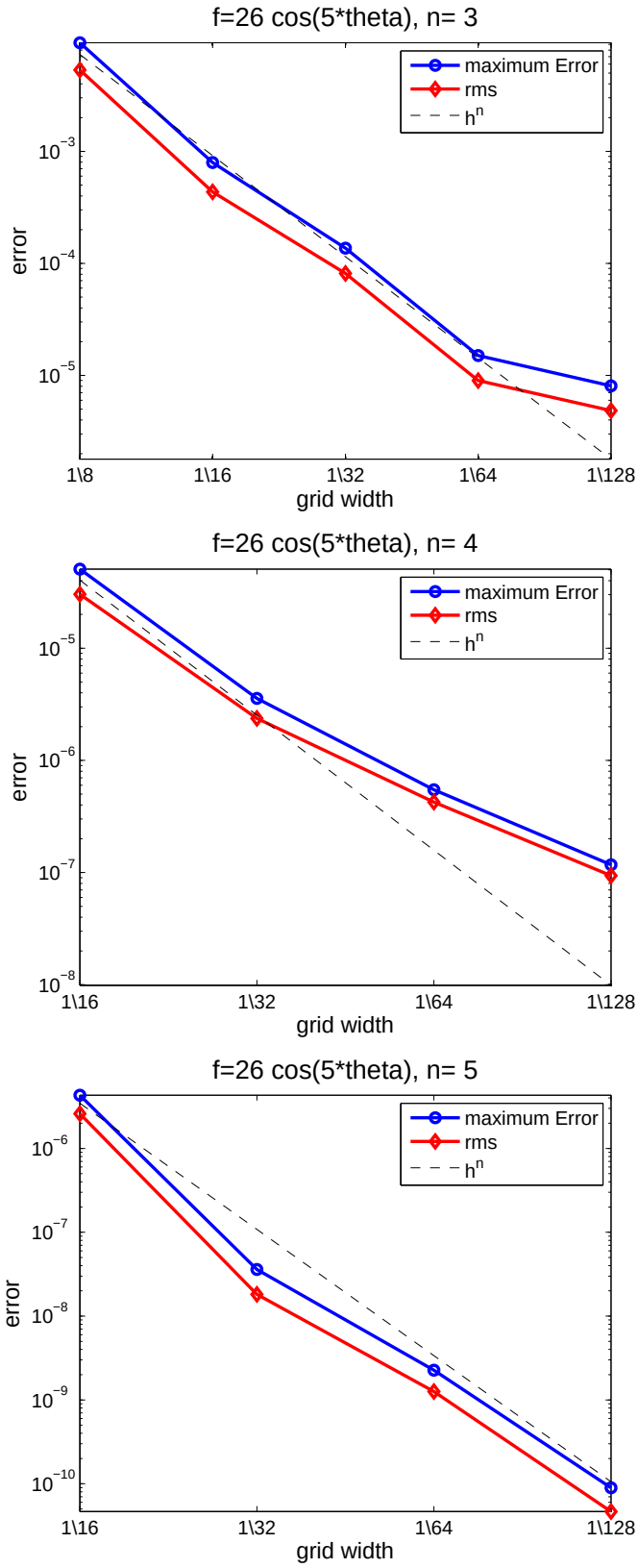

Figure 5.8: the error convergence of the Ambient Signed Distance Method

on Ω and obtain:

$$\sum_i \xi_i \left(-\int_\Omega \left(\triangle_Q b_i \right) b_j + c \int_\Omega b_i b_j \right) = \int_\Omega F b_j \qquad \text{for } i,j \in I$$

$$\sum_i \xi_i \int_\Omega \left(\nabla b_i^\top \cdot \nabla \Phi \right) b_j = 0 \qquad \text{for } i,j \in I$$

with I being the index set of all active B-splines and ξ the control points of U. Unlike before the numbers of equations and variables do not agree. Instead, the number of equations is twice the number of variables. We need to solve an overdetermined linear system of equations. So, the system of linear equations results in the following form:

$$\begin{pmatrix} \text{S+M} \\ \tilde{T} \end{pmatrix} \begin{bmatrix} \\ \xi \\ \\ \end{bmatrix} = \begin{bmatrix} \text{r} \\ \mathbf{0} \end{bmatrix}$$

with stiffness matrix

$$S_{ij} = \int_\Omega (\triangle_Q b_i) b_j,$$

mass matrix

$$M_{ij} = \int_\Omega b_i b_j,$$

outer term matrix

$$T_{ij} = \int_\Omega \left(\nabla b_j^\top \cdot \nabla \Phi \right) b_i,$$

right hand side

$$r_i = \int_\Omega F b_i,$$

ξ is a vector of control points and $\mathbf{0}$ is a zero vector of the same length as r. As in Section 5.6, $\tilde{T}$ is the normalised outer term.

We cannot simplify the term $\left(\triangle_Q b_i \right) b_j$ using partial integration because the active B-splines do not necessarily vanish at the boundary of Ω. We have

$$\begin{aligned} \triangle_Q b_i &= \nabla_Q \cdot \nabla_Q b_i \\ &= Q * (D\, \nabla_Q b_i) \\ &= Q * (D\, (Q \nabla b_i)). \end{aligned}$$

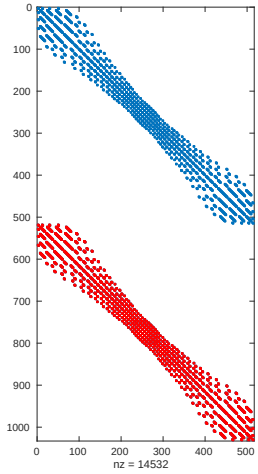

Figure 5.9: the matrix for a 2-D example

So, the stiffness matrix equals

$$S_{ij} = \int_{\Omega} Q * (D\,(Q \nabla b_i)) b_j.$$

The implementation works similar to the description in Section 5.5. This overdetermined linear system of equations can then be solved using the backslash operator in MATLAB. Here the backslash operator solves the linear least squares problem $((S + M + \tilde{T})\xi - r)^2 \rightarrow \min$ using the QR-decomposition. Obviously, $(S + M + \tilde{T})$ is still a sparse matrix. Figure 5.9 depicts the sparsity pattern of the matrix for ω being the unit circle, order $n = 4$ and $h = \frac{1}{16}$.

We tested this modification for the same problem as in Section 5.6, i.e., we solved the intrinsic PDE

$$-\triangle_{\omega} u + u = 26\ \cos(5\theta)$$

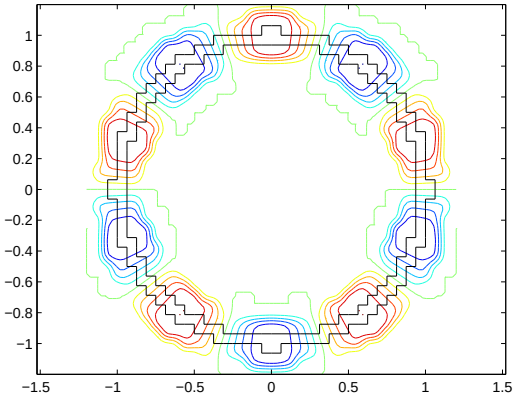

Figure 5.10: contour lines for a 2-D example

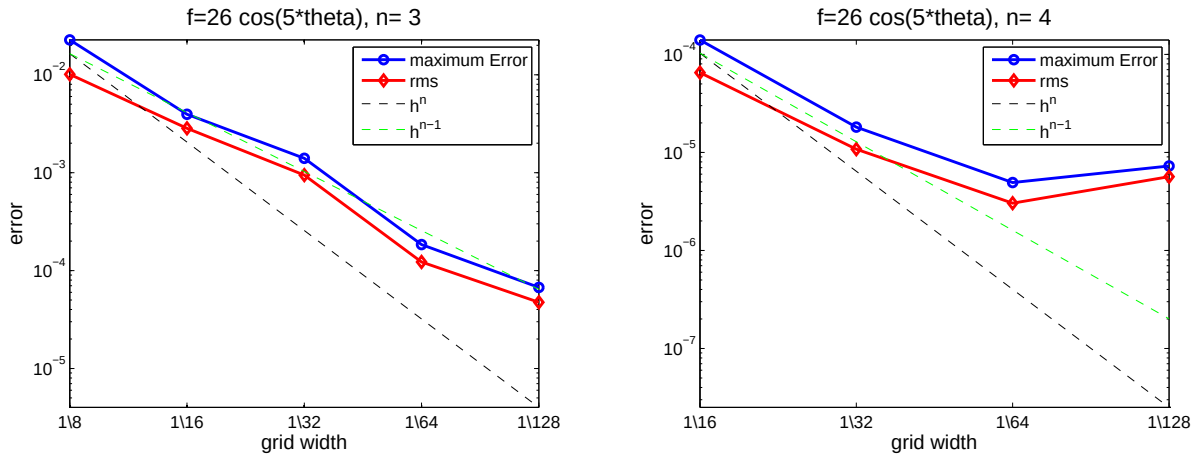

Figure 5.11: the error convergence of the modification

on the unit circle ω. We test the method for the orders $n = 3, 4$ and 5 while the grid width goes from $h = \frac{1}{8}$ to $\frac{1}{128}$. Figure 5.10 depicts the contour lines of the result. We can see that the modification results in a spline that is approximately constant in normal direction inside Ω. The error is computed at about 400 points p_i on ω. The optimal order of convergence $\mathcal{O}(h^n)$ is not given, as shown in Figure 5.11. Here we can see the maximum as well as the root mean square error of the result for different grid widths. For $n = 3$ the error converges in $\mathcal{O}(h^2)$. For greater order n the order of convergence becomes even worse as shown in the right figure for order $n = 4$.

We can conclude that the modification of the Ambient Signed Distance Method for solving PDEs provides an alternative. The ambient domain consists of less cells. This saves calculating time and memory. It results in an approximation of the solution of the given PDE and finishes within less calculation time than the first presented version. Yet, it has some disadvantages: First of all, the unhandy term $D(\nabla_Q b_i)$ needs to be evaluated. Since the second derivative of the B-splines are needed we cannot use the method for a linear approximation. Apart from all, the results do not converge to the exact solution in $\mathcal{O}(h^n)$. Yet, it converges in $\mathcal{O}(h^2)$ for quadratic B-splines on the tested 2-dimensional problem. This is better than presently known methods which mainly use linear approximation and converge in $\mathcal{O}(h^2)$.

6 Ambient Level Set Method

The Ambient Signed Distance Method as presented in Chapter 5 solves given intrinsic linear, elliptic, second-order PDEs on compact submanifolds. It translates the PDE into an embedding PDE on the extended ambient domain of the submanifold. Boundary conditions are formulated in a new way. Finally, the problem is solved with the Finite Element Method. The Ambient Signed Distance Method works for any smooth and compact submanifold of codimension one, regardless of the dimension and genus. It makes use of the signed distance function of the submanifold. If the signed distance function is not given it can be calculated numerically. Yet, this can provide an additional error. A method that avoids the signed distance function is therefore of special interest.

We will now present a method that solves PDEs on submanifolds without using the signed distance function. The method is based on a more general setting and works with any level set function. This gives the method its name: Ambient Level Set Method. The basic idea of this method is the same as in the Ambient B-spline Method (Chapter 4) and the Ambient Signed Distance Method (Chapter 5): The given PDE is extended into some ambient domain of the submanifold. There, it is solved with well known techniques on the Cartesian coordinates. The solution is restricted to the submanifold again where it coincides with the solution of the original problem (see Figure 4.1).

This new method differs from the Ambient Signed Distance Method in the way the functions are extended and how the embedding PDE is constructed. The submanifold is implicitly given as a level set of a smooth function. We want to extend an intrinsic PDE given on this submanifold in such a way that both the solutions coincide on the submanifold and, at the same time, the solution of the embedding PDE is constant along the normals of the level sets. Therefore, the extension operator E that extends functions given on the submanifold to functions defined on the ambient domain has to be chosen differently. In the previous chapter we defined E such that extended functions are constant in normal direction. Now, we want the extensions to be constant along trajectories that are normal on each level set of the level set function.

The construction of the embedding PDE also varies from the previously presented method. Again we limit to linear elliptic second-orders PDEs and concentrate on the model PDE (see Chapter 2). Instead of modifying the intrinsic gradient and the Laplace-Beltrami operator we construct a matrix $A(x,t) \in \mathbb{R}^{d\times d}$, a vector $B(x,t) \in \mathbb{R}^d$ and a constant $C \in \mathbb{R}$ with x on the submanifold $\omega \subset \mathbb{R}^d$ and $t \in \mathbb{R}$ such that the linear differential operator $L(x,u,\nabla_\omega u, \triangle_\omega u)$ can be translated into

$$A * \nabla^2 U + B \cdot \nabla U + C\, U.$$

With help of the level set function we construct $A(x,t)$ and $B(x,t)$ by solving ordinary differential equations. Boundary conditions are added in the same way as for the Ambient Signed Distance Method. The embedding PDE is then solved with the Finite Element Method using tensor product B-splines based on a uniform grid. This procedure is very similar to the technique described in sections 5.3 and 5.4. Finally, we end up in a linear system of equations. We solve it for the control points of the spline. A restriction of the spline to the submanifold approximates the solution of the given intrinsic PDE.

Some notations and the derivation are explained in Section 6.1. Here the basic idea becomes clear. Section 6.2 describes the detailed workflow of the method. We repeat some already mentioned techniques from the previous chapter. An implementation in MATLAB is described in Section 6.3and finally show some numerical results in Section 6.4.

6.1 Derivation

Before we describe the method in detail in the next section we introduce a few notations and the basic idea that leads to the concept of the Ambient Level Set Method. Here, we give importance to the construction of the embedding PDE.

Let $\varphi\colon \mathbb{R}^d \to \mathbb{R}$ with $d \in \mathbb{N}$ be a smooth function with $|\nabla\varphi| \neq 0$ and let ω be implicitly given as a level set of φ such that ω is a compact smooth submanifold of codimension one embedded in $\mathbb{R}^d$. W.l.o.g. we assume ω to be the zero level set of φ, i.e.,

$$\omega = \varphi^{-1}(0)\,.$$

Let G be a grid with grid width $h > 0$. As in Definition 12 we define the extended ambient domain $\bar{\Omega}$ as the union of the support of all active B-splines.

Let $\psi\colon \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ be a flow that is perpendicular to the level sets of φ with $\psi(X,0) = X$ for $X \in \mathbb{R}^d$. So

$$\frac{\partial\psi}{\partial t}(X,t) = \zeta(X)\,\nabla\varphi(\psi(X,t))\,.$$

Here, $\zeta(X)$ can be chosen arbitrarily. We set $\zeta(X)$ as the constant 1. Then,

$$(\psi(X,t))_t = \nabla\varphi(\psi(X,t)).$$

See Figure 6.1. With help of this flow we can now define base points.

Definition 13. *Let ω, $\bar{\Omega}$, and ψ be as above. We call $x_0 \in \omega$ a* base point of $X_0 \in \bar{\Omega}$ with respect to the flow ψ *if there exists $t \in \mathbb{R}$ such that*

$$X_0 = \psi(x_0, t).$$

The map $\mathrm{bp}\colon \bar{\Omega} \to \omega$ returns a base point with respect to ψ:

$$\begin{aligned}&\mathrm{bp}\colon \bar{\Omega} \to \omega\\ &\mathrm{bp}(X) = x \quad \text{such that } \exists t \in \mathbb{R} \text{ with } \psi(x,t) = X \text{ and } \forall \tilde{t} \in [0,t]\ \psi(x,\tilde{t}) \in \bar{\Omega}\,.\end{aligned}$$

In the following let $h > 0$ being small enough such that each point $X \in \bar{\Omega}$ has exactly one base point with respect to ψ and therefore the map $\mathrm{bp} : \bar{\Omega} \to \omega$ is well defined.

The solution of the intrinsic PDE is denoted by u and U is the solution of the embedding PDE. The resulting spline of order $n \in \mathbb{N}$ is denoted by U_h, where h is the grid width. Capital letters are usually used for points in the extended ambient domain $\bar{\Omega}$ and small letters denote points on the submanifold itself. Moreover, derivatives are indicated by subscripts, while superscripts denote vector components.

Until now the extension operator E extended function constantly in normal direction. From now on the operator $E\colon L_2(\omega) \to L_2(\bar{\Omega})$ is defined as follows:

$$(Eu)(X) = u(\mathrm{bp}(X))\,.$$

This means that the function $Eu : \bar{\Omega} \to \mathbb{R}$ is constant along each trajectory $\psi(x,t)$ with $x \in \omega$ fixed.

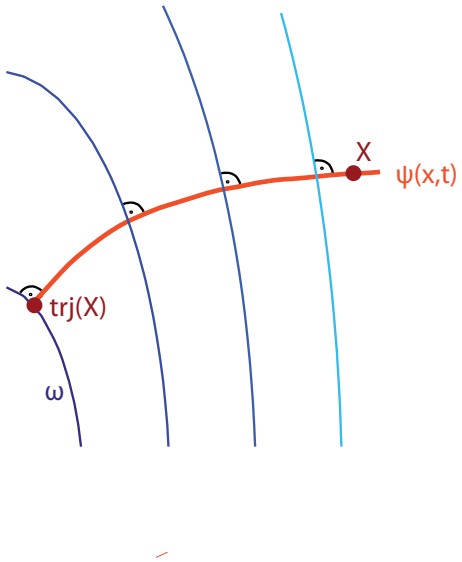

Figure 6.1: a trajectory of the flow ψ

Now, we will describe how to translate the given intrinsic PDE into an embedding PDE on $\bar{\Omega}$. As against the previous method we avoid the signed distance function, but merely make use of the level set function φ. The solution U of the embedding PDE should be constant along the flow ψ, that means $U = Eu$. We therefore have

$$\nabla U^{\top}(X) \cdot \nabla \varphi(X) = 0 \quad \text{for all } X \in \bar{\Omega}$$

as well as

$$U(\psi(X_0, t)) = U(X_0) \quad \text{for any fixed } X_0 \in \bar{\Omega}.$$

Obviously, U should coincide with the solution u of the original problem when restricted to the submanifold ω:

$$U(x) = u(x) \quad \text{for } x \in \omega.$$

Let ν be defined as

$$\nu(X) := \frac{\nabla \varphi(X)}{|\nabla \varphi(X)|}, \quad X \in \bar{\Omega}.$$

We can now compare the intrinsic gradient of u with the gradient of U at points x on the submanifold ω.

Lemma 10. *Let $\omega, \bar{\Omega}$, φ, and E be as above. Let $u \in W_2^1(\omega)$ and $U = Eu$ its extension. Then*

$$\nabla_\omega u(x) = \nabla U(x), \quad x \in \omega.$$

Proof. The intrinsic gradient of u equals the projected gradient of U (see Chapter 2) on the submanifold,

$$\nabla_\omega u(x) = \nabla_P U(x) = (\mathbb{1} - \nu(x) \otimes \nu(x)) \nabla U(x) \quad \text{for } x \in \omega\,.$$

Using $\nabla \varphi(X) \cdot \nabla U(X) = 0$ we have for $x \in \omega$

$$\nabla_\omega u(x) = (\mathbb{1} - \nu(x) \otimes \nu(x)) \nabla U(x) = \nabla U(x) - \frac{\nabla \varphi(x) \cdot \nabla U(x)}{|\nabla \varphi(x)|^2} \nabla \varphi(x) = \nabla U(x).$$

□

The following lemma gives an alternative formulation of the projected divergence operator. This lemma is given and proven in [DDEH10]. We repeat it here for completeness.

Lemma 11. *Let $V : \bar{\Omega} \to \mathbb{R}^d$ be a differentiable vector field such that $V \cdot \nu = 0$. Then*

$$\nabla_P \cdot V(X) = \frac{1}{|\nabla\varphi(X)|} \nabla \cdot \big(V(X)\,|\nabla\varphi(X)|\big) \quad \text{for } X \in \bar{\Omega}.$$

Proof. We drop the arguments for better readability. We have

$$\frac{1}{|\nabla\varphi|} \nabla \cdot \big(V\,|\nabla\varphi|\big) = \sum_{i=1}^{d} \left(V_i^i + \sum_{j=1}^{d} V^i \frac{\varphi_j \varphi_{j,i}}{|\nabla\varphi|^2} \right) = \sum_{i=1}^{d} \left(V_i^i + \sum_{j=1}^{d} V^i \nu^j \frac{\varphi_{i,j}}{|\nabla\varphi|} \right).$$

Differentiating $V \cdot \nu = 0$ we get $\sum_i V_j^i \nu^i + V^i \nu_j^i = 0$ for $1 \le j \le d$. Then

$$\sum_{i=1}^{d} V_j^i \nu^i = -\sum_{i=1}^{d} V^i \nu_j^i = -\sum_{i=1}^{d} V^i \frac{\varphi_{i,j}}{|\nabla\varphi|}, \quad 1 \le j \le d,$$

which implies

$$\frac{1}{|\nabla\varphi|} \nabla \cdot \big(V\,|\nabla\varphi|\big) = \sum_{i=1}^{d} \left(V_i^i - \sum_{j=1}^{d} V_j^i \nu^i \nu^j \right) = \sum_{i=1}^{d} \nabla_P^i V^i = \nabla_P \cdot V.$$

□

With the help of the previous lemma we obtain for the Laplace-Beltrami operator

$$\triangle_\omega u(x) = \triangle_P u(x) = \frac{1}{|\nabla\varphi(x)|} \nabla \cdot (\nabla U(x)\,|\nabla\varphi(x)|) = \triangle U(x) + \frac{\nu(x)^\top H(x)}{|\nabla\varphi(x)|} \nabla U(x), \tag{6.1}$$

where $H = \nabla^2\varphi$ is the Hessian matrix of φ. The first equation is verified in [DE13].

Now, we construct the embedding PDE. The embedding PDE shall be of the form

$$A(x,t) * \nabla^2 U(X) + B(x,t) \cdot \nabla U(X) + C U(X) = E f(X)$$

with $x \in \omega$, $X \in \bar{\Omega}$, $t \in \mathbb{R}$, $f \in L_2(\omega)$, $A(x,t) \in \mathbb{R}^{d\times d}$, $B(x,t) \in \mathbb{R}^d$, and a constant $C \in \mathbb{R}$. Lemma 10 and (6.1) show how to obtain the values of $A(x,0)$ and $B(x,0)$. These values depend on the given intrinsic PDE. For the linear PDE

$$\alpha \triangle_\omega u + \beta \nabla_\omega u + \gamma u = f$$

with $\alpha, \beta, \gamma \in \mathbb{R}$ and some function $f \in L_2(\omega)$ we have

$$A(x,0) = \alpha \cdot \mathbb{1}, \quad B(x,0) = \alpha \frac{\nu^\top H}{|\nabla\varphi|} + \beta\, \mathbf{1} \quad \text{and} \quad C = \gamma$$

for $\mathbf{1}$ be the one-vector. For the elliptic model PDE $-\triangle_\omega u + cu = f$ we have

$$A(x,0) = -\mathbb{1}, \quad B(x,0) = -\frac{\nu^\top H}{|\nabla\varphi|} \quad \text{and} \quad C = c\,.$$

Now, we will look for relationships between differential operators on other level sets and the corresponding intrinsic differential operators. That means, we analyse how $\nabla U(X)$ is related to $\nabla_\omega u(\mathrm{bp}(X))$ and how the Laplacian of U at arbitrary points $X \in \bar{\Omega} \setminus \omega$ is related to the Laplace-Beltrami operator of u at $\mathrm{bp}(X)$.

The flow ψ is orthogonal on the level sets of φ. So

$$(\psi(X,t))_t = \nabla\varphi(\psi(X,t))\,. \tag{6.2}$$

Since U is constant along ψ, i.e., $U(\psi(X,t)) = U(\psi(X,0)) = U(X)$ for any $X \in \bar{\Omega}$ and $t \in \mathbb{R}$ we obtain

$$\sum_{\ell=1}^{d} U_\ell(\psi(X,t))\,\psi_i^\ell(X,t) = U_i(X). \tag{6.3}$$

We drop the arguments and denote it shortly as $\sum_\ell U_\ell \psi_i^\ell = U_i$. The right hand side of this equation is independent of t. We form the derivative of (6.3) with respect to t to obtain

$$\sum_{\ell,m} U_{\ell m}\psi_t^m\psi_i^\ell + U_\ell\psi_{i,t}^\ell = 0\,.$$

Moreover, we find

$$\psi_{j,t}^i = H^{ij}, \quad 1 \le i,j \le d\,. \tag{6.4}$$

We now consider $t = 0$. Then, $\psi(X,0)$ equals the identity map regarding $X \in \bar{\Omega}$ and we have $\psi_i^j = \delta_{ij}$. Together with (6.2) and (6.4) we obtain at $t = 0$

$$\sum_\ell U_\ell H^{\ell i} = -\sum_m U_{im}\nabla\varphi^m\,.$$

We take the derivative with respect to x_j, $1 \le j \le d$, and obtain together with (6.2) and (6.4):

$$\sum_m U_{ijm}\nabla\varphi^m = -\sum_m U_{im}H^{mj} - \sum_\ell U_{\ell j}H^{\ell i} - \sum_\ell U_\ell H_j^{\ell i}\,.$$

So far we have done some precalculations. Now, we will use these equations to compute $A(x,t) \in \mathbb{R}^{d\times d}$ and $B(x,t) \in \mathbb{R}^d$. The embedding PDE should be in such a way that its solution U is constant along the flow ψ. The term

$$\sum_{i,j=1}^{d} A^{ij}(x,t)\,U_{ij}(\psi(x,t)) + \sum_{i=1}^{d} B^i(x,t)\,U_i(\psi(x,t))$$

should be independent of t.

To obtain A and B we do the following computation. Here, we use the previously derived equations and the symmetry of H. Moreover, we use

$$H_j^{\ell i} = \partial_j\psi_{\ell,t}^i = \partial_\ell\psi_{j,t}^i = H_\ell^{ji}\,.$$

For better readability we omit the summation sign and make the convention that we sum up over each index that appears twice:

$$\begin{aligned}
0 &= A_t^{ij} U_{ij} + A^{ij} U_{ijk} \psi_t^k + B_t^i U_i + B^i U_{ik} \psi_t^k \\
&= A_t^{ij} U_{ij} + A^{ij} U_{ijk} \nabla \varphi^k + B_t^i U_i + B^i U_{ik} \nabla \varphi^k \\
&= A_t^{ij} U_{ij} + A^{ij} \left(-U_{i\ell} H^{\ell j} - U_{\ell j} H^{\ell i} - U_\ell H_j^{\ell i} \right) + B_t^i U_i - B^i U_\ell H^{\ell i} \\
&= A_t^{ij} U_{ij} + A^{ij} \left(-U_{i\ell} H^{j\ell} - U_{\ell j} H^{\ell i} - U_\ell H_\ell^{ji} \right) + B_t^i U_i - B^i U_\ell H^{\ell i} \\
&= U_{ij} \left(A_t^{ij} - \left(A^{i\ell} H^{\ell j} + A^{\ell j} H^{i\ell} \right) \right) + U_i \left(B_t^i - \left(B^\ell H^{i\ell} + A^{\ell j} H_i^{\ell j} \right) \right) .
\end{aligned}$$

This equation needs to be fulfilled for any U. One possibility is to set the terms inside the brackets to zero, i.e.,

$$A_t^{ij} - \left(A^{i\ell} H^{\ell j} + A^{\ell j} H^{i\ell} \right) = B_t^i - \left(B^\ell H^{i\ell} + A^{\ell j} H_i^{\ell j} \right) = 0.$$

Then the equation is fulfilled. Therefore, one possible solution for A and B is to choose them as the solutions of the differential equations

$$\begin{aligned}
&A_t(x,t) = A(x,t)H(x,t) + H(x,t)A(x,t), \quad A_0(x) = A(x,0) \quad \text{and} \\
&B_t(x,t) = H(x,t)B(x,t) + A(x,t) * \partial H(x,t), \quad B_0(x) = B(x,0).
\end{aligned} \tag{6.5}$$

The term $A * \partial H$ has to be understood as

$$A * \partial H = (A * \partial_j H)_{j=1,\dots,d} = \begin{pmatrix} A * \partial_1 H \\ \vdots \\ A * \partial_d H \end{pmatrix}.$$

See Definition 11 for the correct interpretation of $*$.

We summarise the result of this chapter in the following theorem.

Theorem 14. *Let ω, $\bar{\Omega}$, φ, H, ψ, and E be given as above. If $U : \bar{\Omega} \to \mathbb{R}$ is twice differentiable, constant along the normal on each level set of φ, and solves*

$$A(x,t) * \nabla^2 U(X) + B(x,t) \cdot \nabla U(X) + c_3 U(X) = Ef(X)$$

with $x \in \omega, t \in \mathbb{R}$ and $X = \psi(x,t)$ where $A(x,t) \in \mathbb{R}^{d \times d}$ solves

$$A_t = AH + HA, \quad A_0 = c_1$$

and $B \in \mathbb{R}^d$ solves

$$B_t = HB + A * \partial H, \quad B_0 = c_1 \frac{\nu^\top H}{|\varphi|} + c_2$$

then the restriction $u = U|_\omega$ solves

$$c_1 \triangle_\omega u(x) + c_2 \nabla_\omega u(x) + c_3 u(x) = f(x).$$

We will consider a few examples to get a better understanding of how to derive $A(x,t)$ and $B(x,t)$.

Example 2. *Let ω be the unit circle embedded in $\mathbb{R}^2$ with some domain $\bar{\Omega} \subset \mathbb{R}^2 \setminus [0,0]$. We use the level set function*

$$\varphi(X,Y) = X^2 + Y^2 - 1\,.$$

The map $\mathrm{bp}\colon \bar{\Omega} \to \omega$ *is defined as*

$$\mathrm{bp}\left(\begin{pmatrix} X \\ Y \end{pmatrix}\right) = \frac{1}{\sqrt{X^2+Y^2}} \begin{pmatrix} X \\ Y \end{pmatrix}\,.$$

Then the Hessian matrix of φ is given as

$$H = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$

and the flow $\psi\colon \bar{\Omega} \times \mathbb{R} \to \mathbb{R}^2$ is

$$\psi([X,Y],t) = \begin{pmatrix} X\, e^{2t} \\ Y\, e^{2t} \end{pmatrix}\,.$$

Now, A and B need to solve the differential equation

$$A_t = AH + HA, \quad \text{and} \quad B_t = HB + A * \partial H$$

for some arbitrary starting values

$$A_0 = \begin{pmatrix} \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_3 \end{pmatrix} \quad \text{and} \quad B_0 = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}\,.$$

We note that A is symmetric. Here, $\partial_x H$ as well as $\partial_y H$ vanish. Therefore,

$$A = \begin{pmatrix} \alpha_1\, e^{4t} & \alpha_2\, e^{4t} \\ \alpha_2\, e^{4t} & \alpha_3 e^{4t} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \beta_1\, e^{2t} \\ \beta_2\, e^{2t} \end{pmatrix}\,.$$

Now we define the function $\tilde{U}\colon \bar{\Omega} \setminus \{(X,Y) \mid X = 0\} \to \mathbb{R}$, $\tilde{U}(X,Y) = \frac{Y}{X}$. Here, $\tilde{U}$ is not the solution of a given PDE but any function that is constant along the flow ψ. We verify that $A * \nabla^2 \tilde{U} + B \cdot \nabla \tilde{U}$ is independent of t:

$$\begin{aligned}
&A * \nabla^2 \tilde{U} + B \cdot \nabla \tilde{U} \\
&= e^{4t} \left(\alpha_1\, \tilde{U}_{XX}(X,\,Y) \,+\, 2\alpha_2\, \tilde{U}_{XY}(X,\,Y) \,+\, \alpha_3\, \tilde{U}_{YY}(X,\,Y)\right) + e^{2t} \left(\beta_1\, \tilde{U}_X(X,\,Y) \,+\, \beta_2 \tilde{U}_Y(X,\,Y)\right) \\
&= \alpha_1 e^{4t} \cdot \frac{e^{2t} y}{e^{6t} x^3} \,+\, 2\alpha_2 e^{4t} \cdot \frac{-1}{e^{4t} x^2} - \,\beta_1 e^{2t} \frac{e^{2t} y}{e^{4t} x^2} \,+\, \beta_2 e^{2t} \frac{1}{e^{2t} x} \\
&= \alpha_1 \frac{y}{x^3} \,- \alpha_2 \frac{2}{x^2} - \,\beta_1 \frac{y}{x^2} \,+\, \beta_2 \frac{1}{x}\,,
\end{aligned}$$

where $[x,y] = \mathrm{bp}([X,Y])$. Now, let the intrinsic PDE be

$$-\triangle_\omega u + u = f$$

for some smooth function f. The embedding PDE is of the form

$$A * \nabla^2 U + B \cdot \nabla U + U = F$$

for $F = Ef$. We then have $\alpha_1 = \alpha_3 = -1$, $\alpha_2 = 0$, $\beta_1 = -\frac{x}{x^2+y^2}$, and $\beta_2 = -\frac{y}{x^2+y^2}$.

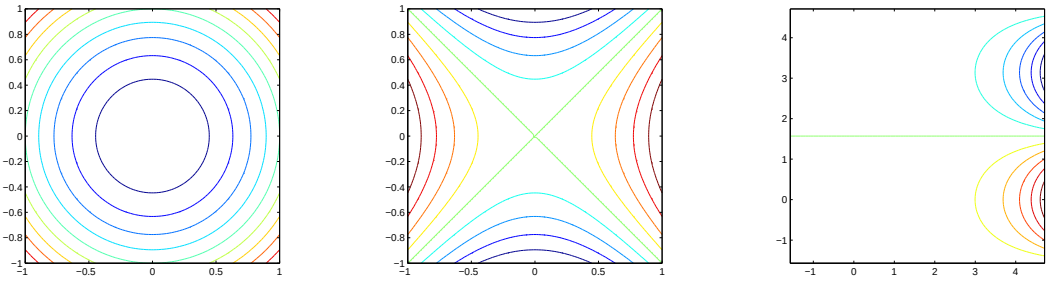

Figure 6.2: level sets of examples 2,3, and 4

We can also apply the presented concept to level set functions that do not have a compact zero level set. The following examples illustrate two more 2-D cases. The level sets to all three examples are depicted in Figure 6.2.

Example 3. *Let* $\varphi(X,Y) = X^2 - Y^2 - \frac{1}{4}$. *Then*

$$\nabla\varphi = \begin{pmatrix} 2X \\ -2Y \end{pmatrix}, \quad H = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix} \quad \text{and} \quad \psi([X,Y],t) = \begin{pmatrix} X\, e^{2t} \\ Y\, e^{-2t} \end{pmatrix}.$$

For A being symmetric we obtain for A and B

$$A = \begin{pmatrix} \alpha_1\, e^{4t} & 0 \\ 0 & \alpha_2\, e^{-4t} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \beta_1\, e^{2t} \\ \beta_2\, e^{-2t} \end{pmatrix}.$$

We again consider the model PDE. Then

$$A_0 = -\mathbb{1} \quad \text{and} \quad B_0 = -\frac{\nu^\top H}{|\nabla\varphi|}$$

and so we get $\alpha_1 = \alpha_2 = -1$, $\beta_1 = -\frac{x}{x^2+y^2}$ *and* $\beta_2 = -\frac{y}{x^2+y^2}$.

Finally, we consider an example where the term ∂H does not vanish.

Example 4. *Let* $\varphi(X,Y) = e^X \cos(Y)$ *and* $H = e^X \begin{pmatrix} \cos(Y) & -\sin(Y) \\ -\sin(Y) & -\cos(Y) \end{pmatrix}$. *We now consider the trajectory* ψ *that moves along the x-axis:*

$$\psi([X,0],t) = \begin{pmatrix} -\ln\left(e^{-X} - t\right) \\ 0 \end{pmatrix}.$$

Next, we compute A and B:

$$A_t = \frac{2}{e^x - t}\begin{pmatrix} A^{11} & 0 \\ 0 & -A^{22} \end{pmatrix}, \qquad A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \Rightarrow \quad A = \begin{pmatrix} \left(\frac{c}{c-t}\right)^2 & 0 \\ 0 & \left(\frac{c-t}{c}\right)^2 \end{pmatrix}$$

$$B_t = \frac{1}{e^x - t}\left(\begin{pmatrix} B^1 \\ -B^2 \end{pmatrix} + \begin{pmatrix} A^{11} - A^{22} \\ 0 \end{pmatrix}\right), \quad B_0 = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad \Rightarrow \quad B = \begin{pmatrix} \frac{\alpha\, c^4 + 2t^2c^2 - \frac{4}{3}t^3c - \alpha t c^3 + \frac{t^4}{3}}{c^2(c-t)^2} \\ \\ \frac{(c-t)\beta}{c} \end{pmatrix}$$

with $c = e^x$.

6.2 Concept of the Method

In the previous section we found a way to translate a given intrinsic PDE into an embedding PDE without using the signed distance function. In this section we will describe the procedure of the complete method. Due to many similarities to Chapter 5 we will not present a detailed discussion but refer to Chapter 5 several times. We concentrate on the model PDE (see Chapter 2)

$$-\triangle_\omega u + cu = f, \quad c > 0,\ f : \omega \to \mathbb{R}$$

on a smooth compact submanifold ω of codimension one that is given as the zero level set of a function $\varphi : \mathbb{R}^d \to \mathbb{R}$ with $|\nabla\varphi| \neq 0$.

Given a grid width $h > 0$ we construct a uniform grid G that covers the whole submanifold. Then we compute all active cells (see Definition 4) and the extended ambient domain $\bar{\Omega}$ as defined in Definition 12. Let I be the index set of all active B-splines.

Next, we formulate the embedding PDE. The embedding PDE is constructed such that its solution U is constant along the normals on the level sets of φ and coincides with the solution u of the original PDE. Let

$$L(x, u(x), \nabla_\omega u(x), \nabla^2_\omega u(x)) = f(x)$$

be the given intrinsic PDE with a linear differential operator L. Then the embedding PDE is of the form

$$A(X) * \nabla^2 U(X) + B(X) \cdot \nabla U(X) + CU(X) = F(X), \quad X \in \bar{\Omega}$$

where $F = Ef$. As derived in the previous section the matrix $A(x,t) \in \mathbb{R}^{d\times d}$ solves the differential equation

$$A_t(x,t) = H(x,t)A(x,t) + A(x,t)H(x,t), \quad A(x,0) = A_0(x)$$

and the vector $B(x,t) \in \mathbb{R}^d$ is a solution of the differential equation

$$B_t(x,t) = H(x,t)B(x,t) + A(x,t) * \partial H(x,t), \quad B(x,0) = B_0(x).$$

The starting values $A_0(x)$ and $B_0(x)$ depend on the intrinsic PDE. We note that the extension operator E extends functions along $\nabla\varphi$.

Now, we divide the extended ambient domain $\bar{\Omega}$ into two parts. The active cells represent the inner part and the remaining cells the outer part (see Figure 5.2 in Chapter 5). On the inner part we solve the embedding PDE and on the outer part we claim

$$\nabla U^\top(X) \cdot \nabla\varphi(X) = 0, \quad X \in \bar{\Omega}.$$

As in the Ambient Signed Distance Method we create the weak formulation of these conditions. For some test function $g : \bar{\Omega} \to \mathbb{R}$ we have on the outer part

$$\int_{\bar{\Omega}} (\nabla U^\top \cdot \nabla\varphi) \cdot g = 0$$

and on the inner part we have

$$\int_{\bar{\Omega}} (A * \nabla^2 U)\, g + \int_{\bar{\Omega}} B \cdot \nabla U \cdot g + C \int_{\bar{\Omega}} Ug = \int_{\bar{\Omega}} Fg\,.$$

We will now solve the given equations with the Finite Element Method. The finite element space is spanned by all relevant tensor product B-splines of order $n \in \mathbb{N}$ based on the grid G. We remind that a B-spline b_i^n is called relevant if $\operatorname{supp} b_i^n \cap \bar{\Omega} \neq \emptyset$ (see Section 5.3). Let I be the index set of the active B-splines and J the index set of the remaining relevant B-splines. For $K = I \cup J$ we have

$$\sum_{k \in K} \xi_k \left(\int_{\bar{\Omega}} (-A * \nabla^2 b_k)\, b_i - \int_{\bar{\Omega}} \left(B \cdot \nabla b_k\right) \cdot b_i + C \int_{\bar{\Omega}} b_k b_i \right) = \int_{\bar{\Omega}} F b_i \qquad \text{for } i \in I$$

$$\sum_{k \in K} \xi_k \int_{\bar{\Omega}} (\nabla b_k^\top \cdot \nabla \varphi) \cdot b_j = 0 \qquad \text{for } j \in J.$$

We combine these equations and write in matrix form

$$(S + cM + T)\, \xi = r$$

with stiffness matrix

$$S_{ij} = \begin{cases} \int_{\bar{\Omega}} (A * \nabla^2 b_j)\, b_i + \int_{\bar{\Omega}} \left(B \cdot \nabla b_j\right) \cdot b_i & \text{if } i \in I \\ 0 & \text{else,} \end{cases}$$

mass matrix

$$M_{ij} = \begin{cases} \int_{\bar{\Omega}} b_i b_j & \text{if } i \in I \\ 0 & \text{else,} \end{cases}$$

outer term matrix

$$T_{ij} = \begin{cases} 0 & \text{if } i \in I \\ \int_{\bar{\Omega}} \left(\nabla b_j^\top \cdot \nabla \varphi\right) b_i & \text{else,} \end{cases}$$

and the right hand side

$$r_i = \begin{cases} \int_{\bar{\Omega}} F b_i & \text{if } i \in I \\ 0 & \text{else.} \end{cases}$$

Compared with the Ambient Signed Distance Method the mass matrix M_{ij} and the outer term matrix T_{ij} remain the same. One needs to keep in mind that now φ is an arbitrary level set function.

To evaluate F at some point $X_0 \in \bar{\Omega}$ an ordinary differential equation has to be solved. We have to source the flow ψ until it reaches the submanifold ω, evaluate f at that point and finally assign that function value to X_0. We have

$$F(X_0) = F(\psi(x_0, t)) = f(\psi(x_0, 0)) = f(x_0) \quad \text{with} \quad x_0 = \operatorname{bp}(X_0).$$

We compute the flow by solving

$$\psi_t(X) = \nabla\varphi(X), \quad \psi_0 = \psi(x_0, 0) = x_0.$$

Then the equation

$$X_0 = \psi(x_0, t_0)$$

is reformulated such that $x_0 \in \omega$ is expressed by a term depending on X_0 and t_0. We insert this term in the equation $\varphi(x_0) = 0$ and compute t_0. There are several methods to find the root of functions. For example, Newton's method can be used on smooth functions. This is a numerical method that recursively finds better approximations of roots of real valued functions. With the value of t_0 we can derive x_0 easily. In this way we can evaluate $F(X) = f(\mathrm{bp}(X))$ at any point in the extended ambient domain and estimate the right hand side r_i.

The stiffness matrix S cannot be adopted from the Ambient Signed Distance Method. Here, we first need to compute $A(x,,t)$ and $B(x,t)$ as described in the previous section. They are solutions of the differential equations (6.5). This can either be done directly or by using a preprogrammed solver. In the examples presented in Section 6.4 we solved them directly. To evaluate $A(x,t)$ and $B(x,t)$ for a given point X we again need $x = \mathrm{bp}(X)$ and t. Therefore, we apply the above mentioned procedure again.

Finally, the linear system of equations is solved for ξ and we obtain a spline

$$U_h = \sum_{k \in K} \xi_k b_k^n.$$

We restrict the spline to the submanifold to obtain the approximation of the exact solution of the intrinsic PDE.

6.3 Implementation

The implementation of this method is done in MATLAB. It shows many similarities to the implementation of the Ambient Signed Distance Method in Section 5.5. The INPUT of this method is

- a manifold ω, given as the zero level set of some function $\varphi : \mathbb{R}^d \to \mathbb{R}$,
- a grid width $h > 0$,
- an order $n \in \mathbb{N}$ and
- a continuous function $f : \omega \to \mathbb{R}$.

Since the second derivative of the B-splines are used in this method we need to limit to orders $n > 2$. The OUTPUT is

- the index set K of all relevant B-splines
- a vector containing the control points ξ_k, $k \in K$ of the resulting spline U_h and
- the maximal as well as the root mean square error of $U_h \in S^n(\bar{\Omega})$.

The computation of the active cells, the extended ambient domain $\bar{\Omega}$ and the relevant B-splines proceeds in the same way as in the implementation of the previously introduced methods. To compute the stiffness matrix

$$S_{ij} = \int_{\Omega} \left(A * \nabla^2 b_j(x) + B \cdot \nabla b_j(x) \right) b_i(x)$$

we need to solve the differential equations (6.5) for $A(x,t)$ and $B(x,t)$. This has to be done only once at the beginning of the calculation. These differential equations can be solved using MAPLE or an built-in method of MATLAB, like for example *ode45*. For the examples presented in the next section we did not use any preprogrammed solver but solved the differential equations directly.

The same holds for the trajectory ψ. The trajectories ψ are needed to evaluate function values of extended functions, i.e., to evaluate $F = Ef$ at points $X \in \bar{\Omega}$.

We implemented the subfunction *[x,t]=basepoint(X)*. Given a point $X \in \bar{\Omega}$ it returns the point $x = \mathrm{bp}(X)$ as well as $t \in \mathbb{R}$ such that $\psi(x,t) = X$. The function works as follows: We express $x \in \omega$ by a term $x(X,t)$ depending on X and t and insert this term in the equation $\varphi(x) = 0$ and solve it for t. This is done by the build-in function *fzero*. The algorithm of this function is a combination of bisection and inverse quadratic interpolation. Knowing the value of t the point $x \in \omega$ can be computed by using the term $x(X,t)$.

After computing $x = \mathrm{bp}(X)$ we can easily assign $F(X) = f(x)$. To compute the integral

$$r_i = \int_{\bar{\Omega}} F \; b_i^n$$

we use Romberg's method (see Section 5.5). The entries of the outer term matrix

$$T_{ij} = \int_{\bar{\Omega}} (\nabla \varphi^{\top} \cdot \nabla b_j) \, b_i$$

are also computed via Romberg's method while the entries of the mass matrix

$$M_{ij} = \int_{\bar{\Omega}} b_j b_i$$

can be computed by Gaussian quadrature. See Section 5.5 for a more detailed description.

For the stiffness matrix we precompute the entries of $A(x,t)$ and $B(x,t)$. Since they depend on $x = \mathrm{bp}(X)$ for a given $X \in \bar{\Omega}$ and $t \in \mathbb{R}$ we need to run the previously described subfunction *[x,t]=basepoint(X)* again. Then for each point $X \in \bar{\Omega}$ the entries α_{ij}, $i,j = 1,\ldots,d$, of A and β_i, $i = 1,\ldots,d$, of B can be computed and used to calculate the integral

$$S_{ij} = \int_{\bar{\Omega}} (A * \nabla^2 b_j + B \, \nabla b_j) b_i = \int_{\bar{\Omega}} \left(\sum_{\ell,k=1}^{d} \alpha_{\ell,k} \frac{\partial^2 b_j}{\partial x_\ell \, \partial x_k} + \sum_{k=1}^{d} \beta_k \frac{\partial b_j}{\partial x_k} \right) b_i .$$

We solve the linear system of equations by using the MATLAB backslash-operator. The program returns the so computed coefficients ξ_k, $k \in K$ where K is the index set of all relevant B-splines.

Finally we compute the error of the result. Therefore, the spline is evaluated at a finite number of points p_i on the submanifold. If the solution u of the given problem is known, we can compare the values and return the maximal as well as the root mean square error. See Section 4.3 for a definition of these. If the solution is unknown we test how good the spline solves the given intrinsic PDE. That means we compute $F(p_i, U_h(p_i), \nabla_\omega U_h(p_i), \nabla^2_\omega U_h(p_i))$ where U_h is the resulting spline of the method. We note that in this case the optimal order of convergence would be in $\mathcal{O}(h^{n-2})$ for second-order PDEs. In the examples presented in the next section we know the exact solution. Therefore, we can compute the error by comparing the resulting spline to the exact solution.

6.4 Numerical Results

In this section we show some numerical results of the method as described in Section 6.2 and implemented in MATLAB as suggested in Section 6.3. We consider three 2-D examples. All test manifolds ω_i, $i = 1, 2, 3$, are closed curves embedded in $\mathbb{R}^2$. On these test manifolds we solve the elliptic PDE

$$-\triangle_\omega u + u = f$$

for some smooth test functions $f : \omega_i \to \mathbb{R}$. We will see how $A(x,t)$ and $B(x,t)$ are computed and how the embedding PDE is built by making use of the flow ψ.

At first we test the case of Example 2. The test manifold ω_1 is the unit circle embedded in $\mathbb{R}^2$. The circle is presented as the zero level set of

$$\varphi_1(X,Y) = X^2 + Y^2 - 1\,.$$

We note that φ_1 is not the signed distance function of the unit circle. In Example 2 we already computed

$$A([x,y],t) = \begin{pmatrix} -e^{4t} & 0 \\ 0 & -e^{4t} \end{pmatrix} \quad \text{and} \quad B([x,y],t) = -\frac{1}{x^2+y^2}\begin{pmatrix} x\,e^{2t} \\ y\,e^{2t} \end{pmatrix}.$$

The embedding PDE is of the form

$$A * \nabla^2 U + B\nabla U + U = F\,.$$

Now, using polar coordinates (r,θ) we define

$$f(r,\theta) = 26\,\cos(5\,\theta).$$

We note that the intrinsic PDE coincides with the one of Section 5.6. There we solved it using the Ambient Signed Distance Method and obtained good results.

For this test manifold ω_1 the subfunction *basepoint* (see Section 6.3) has not to be defined as a numerical method. Instead, we define

$$\text{basepoint}([X,Y]) := \left(\frac{1}{\sqrt{X^2+Y^2}}\begin{pmatrix} X \\ Y \end{pmatrix}, \frac{1}{\sqrt{X^2+Y^2}} - 1 \right).$$

We compute the matrix $S + M + T$ (see Section 6.2) and solve the linear system of equations

$$(S + M + T)\,\xi = r\,.$$

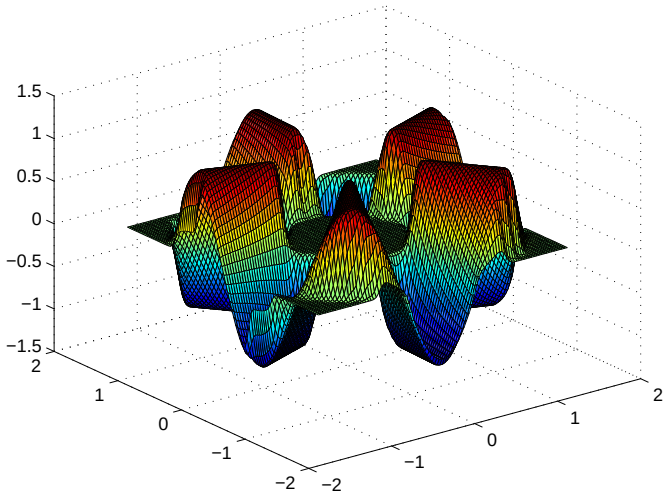

Figure 6.3: the resulting spline

The matrix has the same structure as depicted in Figure 5.4. As we have already seen in the numerical experiments of the Ambient Signed Distance Method, the condition of the matrix becomes very big for $h \to 0$. This is based on the fact that the values in the matrices converge differently with shrinking h. This influences the accuracy of calculation negatively. Therefore, we use a normalised outer term $\tilde{T}$ that divides each row by its diagonal element:

$$\tilde{T}_{ij} = \frac{T_{ij}}{T_{ii}}.$$

Since we know the solution of the intrinsic PDE we can compute the error by comparing the result with the exact solution

$$u = \cos(5\,\theta).$$

Figure 6.3 shows the resulting spline of the Ambient Level Set Method and Figure 6.4 depicts the error. Both pictures look very similar to the corresponding ones of Section 5.6.

Now, let us have a look at the contour lines. These are depicted in Figure 6.5 for $n = 4$ and $h = \frac{1}{16}$. We can see that the contour lines form straight lines and therefore coincide with the flow ψ. So the resulting spline is approximately constant along ψ.

To compute the error we evaluate the spline $U_h = \sum_{k \in K} \xi_k \, b_k^n$ at about 400 equally spread points p_i, $i = 1, \ldots, 400$, on ω_1. Here, K is the index set of all relevant B-splines. We tested the method for the orders $n = 3, 4, 5$ and grid widths $h = 2^{-m}$, $m = 3, \ldots, 7$. The error behaviour is shown in Figure 6.6. The blue lines show the maximal error and the red lines illustrate the root mean square error.

Figure 6.6 shows very good error behaviour of the Ambient Level Set Method for this example. In all three orders $n = 3, 4, 5$ both the errors shrink with the grid width h. The resulting spline converges to the solution u for $h \to 0$.

Since we solved the same problem with the previously presented Ambient Signed Distance Method as well, we can compare both results here. Both methods give approximately equivalent results. Figure 6.7 compares the root mean square error of both methods for $n = 3$ and $n = 5$. In the quadratic case the Ambient Level Set Method gives better results, while in the case of $n = 5$

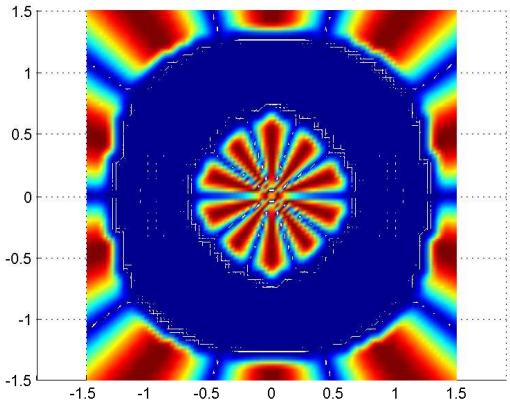

Figure 6.4: error of the result

the Ambient Signed Distance Method shows a better result for small grid widths. In this example the Ambient Level Set Method needs less calculation time than the Ambient Signed Distance Method.

The second test manifold ω_2 is an ellipse as the zero level set of

$$\varphi_2(X,Y) = \alpha X^2 + Y^2 - 1, \quad \alpha \in \mathbb{R}^+.$$

The left picture of Figure 6.8 shows the ellipse ω_2 with different level sets of φ_2 for $\alpha = 16$. The picture on the right side shows the inner and the outer part of the extended ambient domain $\bar{\Omega}_2$ for a grid width $h = \frac{1}{16}$ and order $n = 3$. First of all we compute the flow $\psi : \bar{\Omega}_2 \times \mathbb{R} \to \mathbb{R}^2$. Solving the ordinary differential equation

$$\psi_t = \nabla\varphi_2 = \begin{pmatrix} 2\alpha\, x \\ 2\, y \end{pmatrix}, \quad \psi([X,Y],0) = \begin{pmatrix} X \\ Y \end{pmatrix}$$

we get

$$\psi([X,Y],t) = \begin{pmatrix} X\, e^{2\alpha t} \\ Y\, e^{2t} \end{pmatrix}.$$

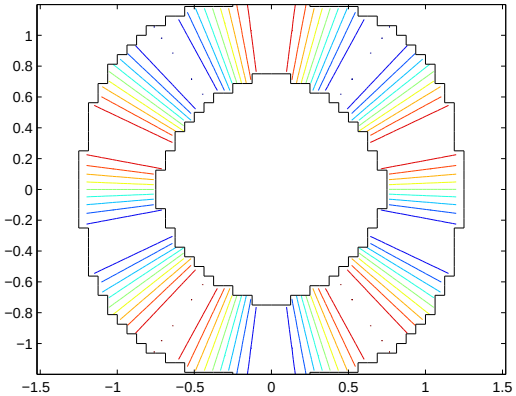

Figure 6.5: contour lines of the spline U_h

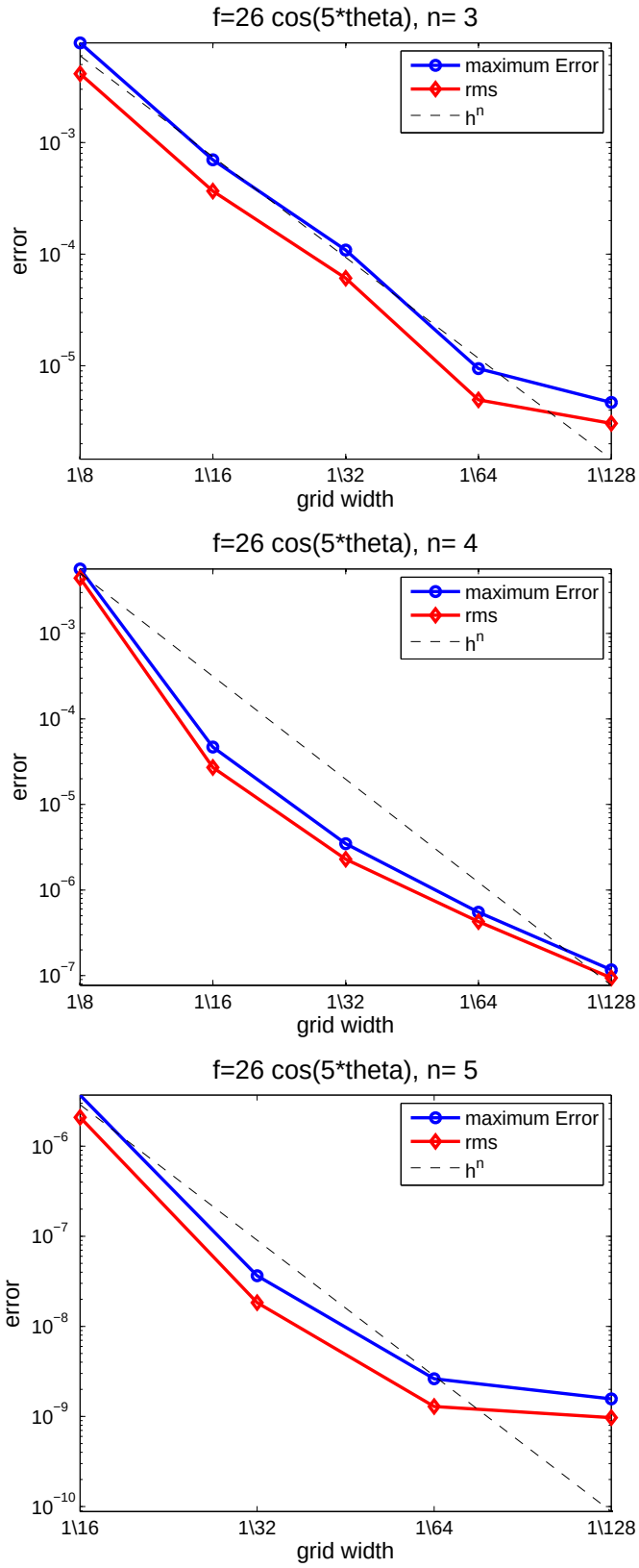

Figure 6.6: experimental rate of convergence on the unit circle

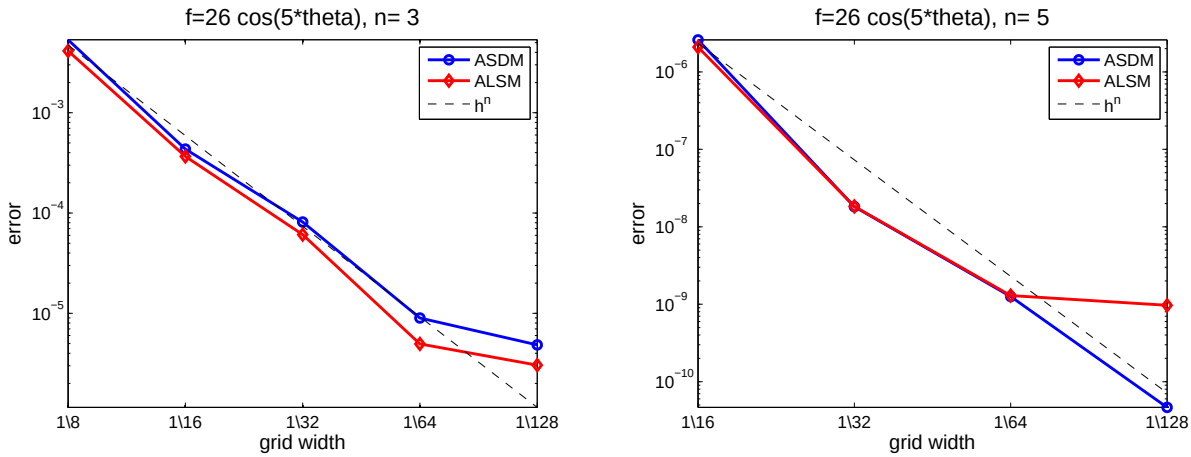

Figure 6.7: comparison of the Ambient Signed Distance Method and the Ambient Level Set Method

Let $[X,Y] \in \bar{\Omega}_2$ be any point in the extended ambient domain of the ellipse. To evaluate $F(X,Y)$ we consider the following:

$$\begin{aligned} & x = X\, e^{-2\alpha t}, \quad y = Y\, e^{-2t} \quad \text{for } [x,y] = \mathrm{bp}([X,Y]) \\ & \alpha x^2 + y^2 - 1 = 0 \\ \Rightarrow \quad & \alpha (X\, e^{-2\alpha t})^2 + (Y\, e^{-2t})^2 = 1. \end{aligned}$$

We solve this equation for t and then compute x and y. With this we can evaluate the extended function $F = Ef$ at any point $[X,Y] \in \bar{\Omega}_2$ by $F(X,Y) = f(\mathrm{bp}(X,Y))$. After we computed the extended function F we do the following calculation to obtain $A([x,y],t)$ and $B([x,y],t)$. The Hessian matrix of the level set function φ_2 takes the form:

$$H = \begin{pmatrix} 2\alpha & 0 \\ 0 & 2 \end{pmatrix}.$$

We have

$$A' = HA + AH, \qquad A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

which implies

$$A([x,y],t) = \begin{pmatrix} -e^{4\alpha t} & 0 \\ 0 & -e^{4t} \end{pmatrix}.$$

For B we solve the differential equation

$$B' = HB + A * \partial H = HB$$

with the starting value

$$B_0 = -\frac{\nu^\top H}{|\nabla \varphi_2|} = \frac{-1}{|\nabla \varphi_2|^2} \begin{pmatrix} 2\alpha x, & 2y \end{pmatrix} \begin{pmatrix} 2\alpha & 0 \\ 0 & 2 \end{pmatrix} = \frac{-1}{\alpha^2 x^2 + y^2} \begin{pmatrix} \alpha^2 x, & y \end{pmatrix}.$$

We get

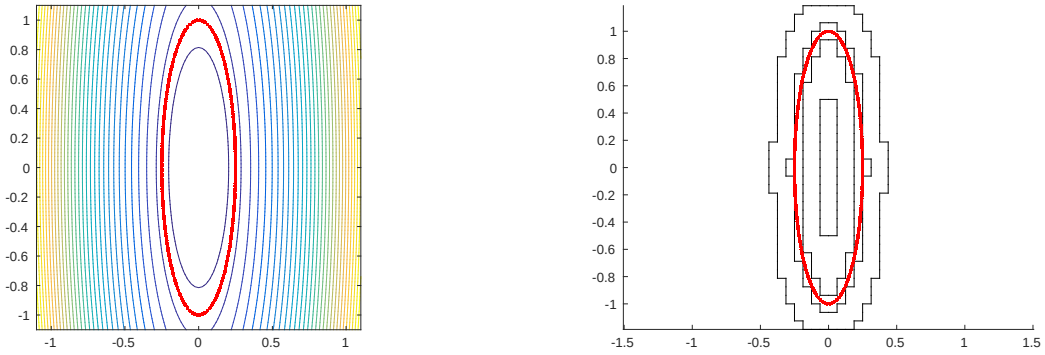

Figure 6.8: *left:* the ellipse ω_2 and different level sets of φ_2, *right:* inner and outer part

$$B([x,y],t) = \frac{-1}{\alpha^2 x^2 + y^2} \begin{pmatrix} \alpha^2 x\, e^{2\alpha t} \\ y\, e^{2t} \end{pmatrix} .$$

With these data we can now solve the linear system of equations

$$(S + M + \tilde{T})\,\xi = r$$

with stiffness matrix S, mass matrix M, normalised outer term $\tilde{T}$ and right hand side r as described in the previous section.

We tried our method on two test functions. The first test function is given as

$$f_1(x,y) = \frac{1024x}{\left(y^2 + 256x^2\right)^2} + 4x\,.$$

The exact solution of the intrinsic PDE $-\triangle_{\omega_2} u + u = f_1$ for $\alpha = 16$ is

$$u(x,y) = 4x\,.$$

The second test function f_2 is such that

$$(f_2 \circ \chi)(t) = \frac{800\,(9\,\sin(3t) + 17\,\sin(5t) + 6\,\sin(7t))}{(15\,\cos(2t) + 17)^2} + \sin(5t)$$

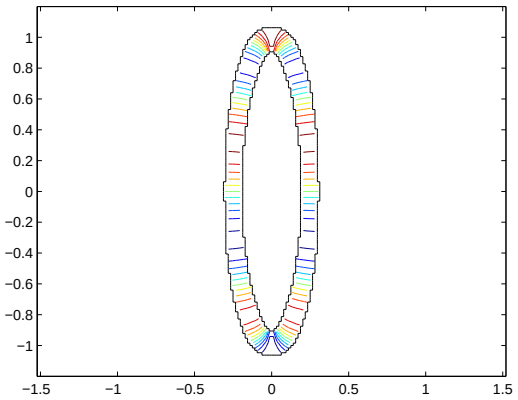

Figure 6.9: contour lines of the spline U_h

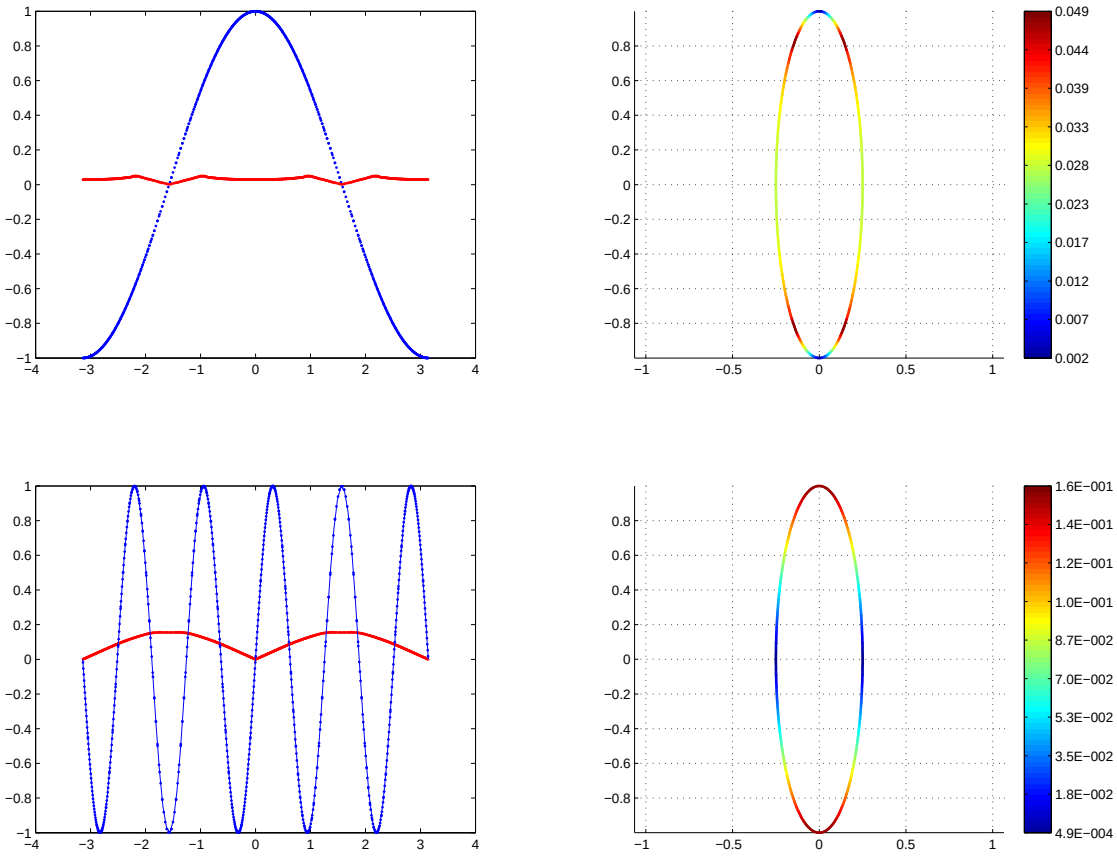

Figure 6.10: error on the ellipse

for the parametrisation $\chi(t) = \left(\frac{1}{4}\cos(t), \sin(t)\right)$ for $t \in [-\pi, \pi)$. Then, the exact solution of the intrinsic PDE is given by

$$(u \circ \chi)(t) = \sin(5\,t)\,.$$

The contour lines of the resulting spline U_h for f_2 with $h = \frac{1}{64}$ and $n = 4$ are shown in Figure 6.9. Here, we can see that they are no straight lines. Instead they are perpendicular to the level sets of Figure 6.8, as required.

We choose around 750 equally spread points on ω_2 and compute the maximal and the root mean square error. In Figure 6.10 the maximal error is depicted for both test functions. Here we used order $n = 4$, $\alpha = 16$, and grid width $h = \frac{1}{32}$ for f_1 and grid width $h = \frac{1}{64}$ for f_2. On the left hand side we see the function value in blue and the error in red. Here, we used the parametrisation χ as given above. The pictures on the right hand side show the error on the ellipse itself. We can see that the error increases with the curvature of the submanifold.

Figure 6.11 depicts the order of convergence for the two test functions f_1 and f_2, $\alpha = 16$, for orders $n = 3$ and 4, and grid widths going from $h = \frac{1}{32}$ to $h = \frac{1}{128}$. For $f = f_1$ the error increases again for very small values of h. In case of second test function the error shrinks with h. Yet, the order of convergence is very low.

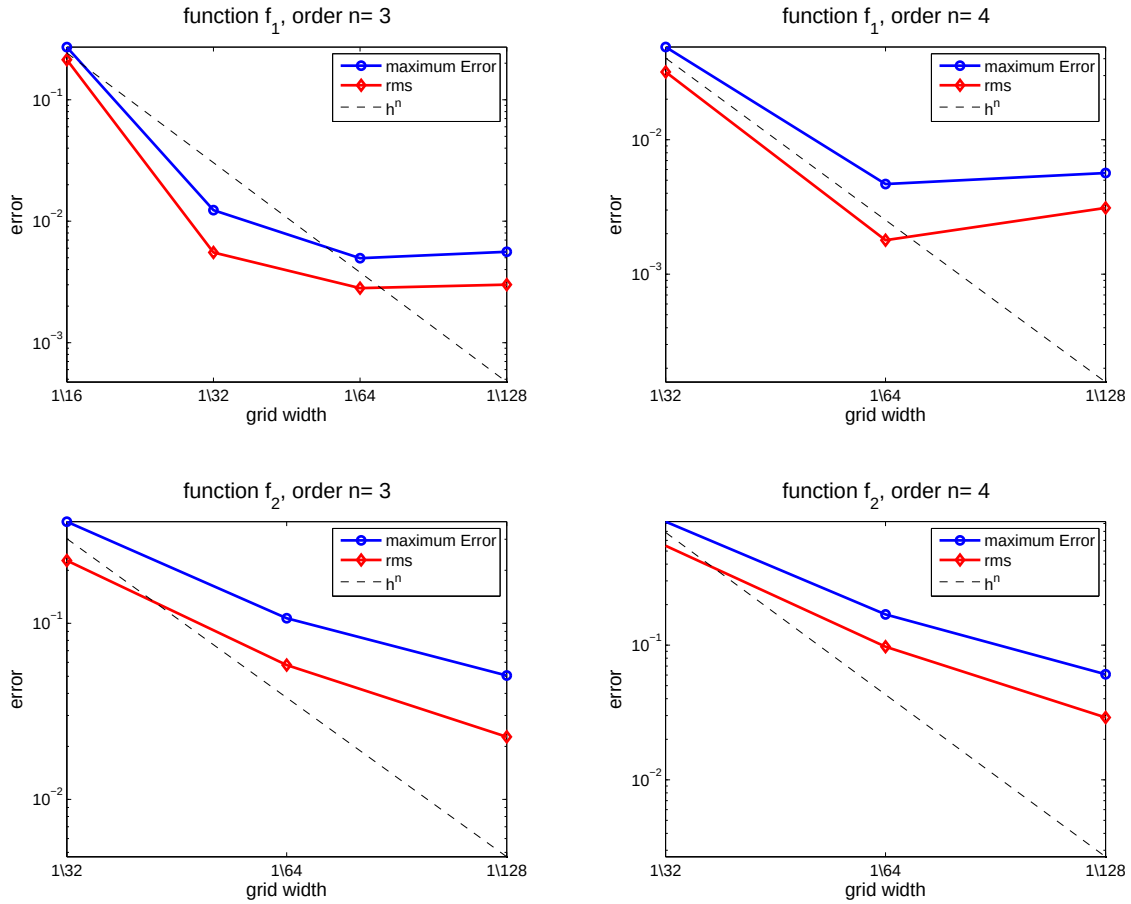

Figure 6.11: error convergence for PDEs on the ellipse

Now we compute one more 2-D example in which ∂H does not vanish. The third test manifold ω_3 is the zero level set of the function

$$\varphi_3(X,Y) = X^2 + Y^4 - 1\,.$$

We have

$$\nabla\varphi_3 = \begin{pmatrix} 2X \\ 4Y^3 \end{pmatrix} \quad \text{and} \quad H = \begin{pmatrix} 2 & 0 \\ 0 & 12Y^2 \end{pmatrix}.$$

While $\partial_X H$ is the zero matrix $\partial_Y H$ does not vanish. We have

$$\partial_Y H = \begin{pmatrix} 0 & 0 \\ 0 & 24Y \end{pmatrix}.$$

To obtain the flow ψ we need to solve the differential equation

$$\psi_t = \begin{pmatrix} 2x \\ 4y^3 \end{pmatrix} \quad \text{with} \quad \psi([X_0,Y_0],0) = \begin{pmatrix} X_0 \\ Y_0 \end{pmatrix}.$$

Solving this we get

$$\psi = \begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix}$$

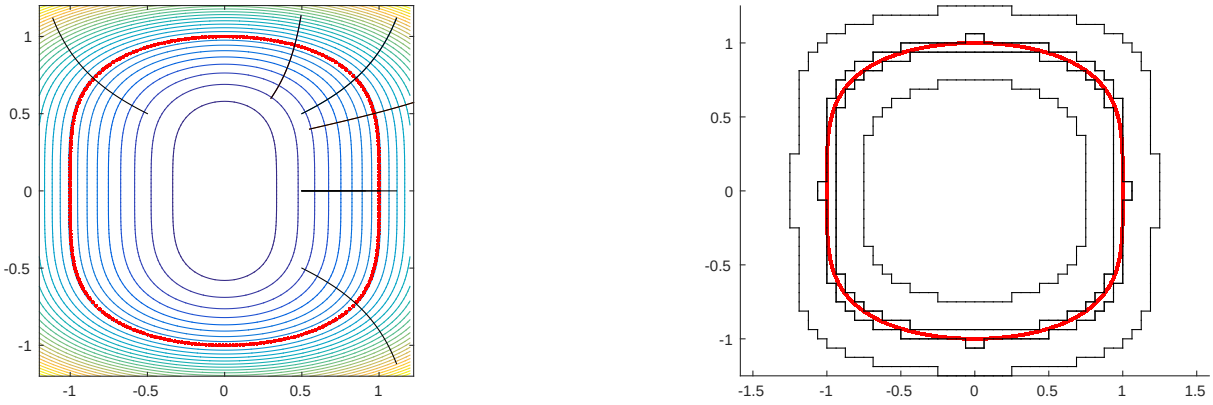

Figure 6.12: *left:* manifold ω_3 and different level sets of φ_3, *right:* inner and outer part of $\bar{\Omega}_3$

with

$$\psi_1 = X_0\, e^{2t} \quad \text{and} \quad \psi_2 = \begin{cases} -\frac{1}{\sqrt{2(c-4t)}} & \text{for } Y_0 < 0 \\ 0 & \text{for } Y_0 = 0\,, \\ \frac{1}{\sqrt{2(c-4t)}} & \text{for } Y_0 > 0 \end{cases} \quad \text{with } c = \frac{1}{2Y_0^2}.$$

Figure 6.12 shows different level sets of φ_3 as well as the zero level set ω_3 in red and a few randomly chosen trajectories $\psi([X,Y],t)$ with $[X,Y]$ fixed. The right picture shows the inner and outer part of the extended ambient domain $\bar{\Omega}_3$ for a grid width $h = \frac{1}{16}$ and order $n = 4$.

First of all, we need to implement the function $\mathrm{bp} : \bar{\Omega}_3 \to \omega_3$ as described in Section 6.3. For any point $[X,Y] \in \bar{\Omega}_3 \setminus \{[X,Y] \in \mathbb{R}^2 \mid Y = 0\}$ we have

$$\begin{aligned} X = x\, e^{2t} \quad &\Rightarrow \quad x = X\, e^{-2t} \\ Y = \frac{\pm 1}{\sqrt{\frac{1}{y^2} - 8t}} \quad &\Rightarrow \quad |y| = \frac{1}{\sqrt{\frac{1}{Y^2} + 8t}} \quad \Rightarrow \quad y = \operatorname{sign}(Y) \frac{1}{\sqrt{\frac{1}{Y^2} + 8t}} \end{aligned}$$

for $[x,y] = \mathrm{bp}([X,Y])$. for $Y = 0$ we have $y = 0$. That means, we compute the zero of

$$\begin{aligned} & \left(X\, e^{-2t}\right)^2 + \left(\frac{1}{\sqrt{\frac{1}{Y^2} + 8t}} \right)^4 - 1 = 0 \\ \Leftrightarrow \quad & X^2\, e^{-4t} + \frac{1}{\left(\frac{1}{Y^2} + 8t\right)^2} - 1 = 0 \end{aligned}$$

to obtain t and then insert that value in the above equations to get x and y. In this way the extension F can be evaluated as $F(X,Y) = f(x,y)$.

To compute $A([x,y],t)$ and $B([x,y],t)$ we proceed as follows. We need to solve the differential equations (6.5) derived in Section 6.1. So, for $A([x,y],t)$ we solve

$$A_t = HA + AH \quad \text{with} \quad A([x,y],0) = \mathbb{1}$$

and obtain

$$A([x,y],t) = \begin{pmatrix} e^{4t} & 0 \\ 0 & \frac{1}{\left(1-8ty^2\right)^3} \end{pmatrix}.$$

We note that $\bar{\Omega}_3$ should be so small that $t < \frac{1}{8}$. For the computation of B we have

$$B_t = HB + A * \partial H = \begin{pmatrix} 2\,B_1 \\ 12\,Y^2 B_2 \end{pmatrix} + \begin{pmatrix} 0 \\ A_{22} \cdot 24\,Y \end{pmatrix} = \begin{pmatrix} 2\,B_1 \\ \frac{12\,B_2}{\frac{1}{y^2}-8t} + \frac{\operatorname{sign}(y)\cdot 24}{y^6\left(\frac{1}{y^2}-8t\right)^{\frac{7}{2}}} \end{pmatrix}.$$

The starting condition for B is

$$B_0 = \frac{\nu^\top H}{|\nabla\varphi|} = \frac{1}{x^2+4y^6}\begin{pmatrix} x \\ 12y^5 \end{pmatrix}.$$

We note that we use the starting condition $A_0 = \mathbb{1}$ and $B_0 = \frac{\nu^\top H}{|\nabla\varphi|}$. Therefore we will use the stiffness matrix $-S$ instead of S. We solve this differential equation and get

$$B_1([x,y],t) = \frac{x}{x^2+4y^6}\,e^{2t}$$

and

$$B_2([x,y],t) = \begin{cases} \frac{-3\left(\frac{1}{y^2}-8t\right)^{\frac{1}{2}}}{(1-8ty^2)^3} + \frac{b_0+\frac{3}{y}}{(1-8ty^2)^{\frac{3}{2}}} & \text{for } y > 0 \\ 0 & \text{for } y = 0 \\ \frac{3\left(\frac{1}{y^2}-8t\right)^{\frac{1}{2}}}{(1-8ty^2)^3} + \frac{b_0-\frac{3}{y}}{(1-8ty^2)^{\frac{3}{2}}} & \text{for } y < 0 \end{cases}$$

with $b_0 = \frac{12y^5}{x^2+4y^6}$. Now, we consider the parametrisation

$$\chi(t) = \left(\cos(t),\, \operatorname{sign}(t)\sqrt{|\sin(t)|}\right) \quad \text{for } t \in [-\pi,\pi)$$

and define the test function

$$f_3(x,y) = \frac{x\,y^4\left(x^2-y^4+5\right)}{2\operatorname{sign}(y)y^3\sqrt{\frac{x^2}{4y^2}+y^4}\left(4y^6+x^2\right)^{\frac{3}{2}}} + x\,.$$

The exact solution of the PDE

$$-\triangle_{\omega_3} u + u = f_3$$

is

$$u(x,y) = x\,.$$

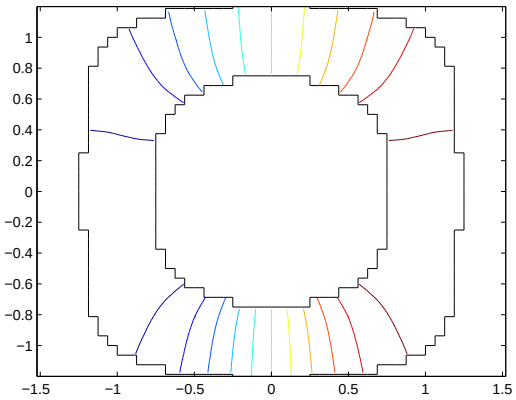

Figure 6.13: contour lines of the resulting spline on $\bar{\Omega}_3$

The contour lines of the resulting spline for $h = \frac{1}{16}$ and $n = 4$ are depicted in Figure 6.13. Again they look perpendicular to the level sets, which is intended. For the error analysis we concentrate on the upper half space $H^+ := \{(X, Y) \in \mathbb{R}^2 \mid Y > 0\}$.

The maximal error on H^+ is depicted in the left illustration of Figure 6.14. We can see that the error stays small inside the extended ambient domain $\bar{\Omega}_3$. There we have a an error in the magnitude of 10^{-6}. The right picture of Figure 6.14 shows the maximal error on $H^+ \cap \omega$. Here, we can see that the error grows with the curvature of the submanifold. We compute about 350 points p_i, $i = 1, \ldots, 350$ on $\omega \cap H^+$. Figure 6.15 depicts the maximal as well as the root mean square error for orders $n = 3, 4$ and 5 and grid width h ranging from $\frac{1}{16}$ to $\frac{1}{128}$. The red lines depict the maximal error and the blue lines the root mean square error. Both axes have a logarithmic scale. We can see that both errors shrink with h. In this example, a higher order n does not improve the result.

We can conclude that the Ambient Level Set Method provides a possibility to solve intrinsic linear elliptic second-order PDEs on smooth compact submanifolds embedded in $\mathbb{R}^d$. The used tools in this method are simple and well-known. Only the way we handle the boundary conditions is new. The method is easy to implement. Since the result does not yet show the wanted order of convergence, there seems to be still room for improvement.

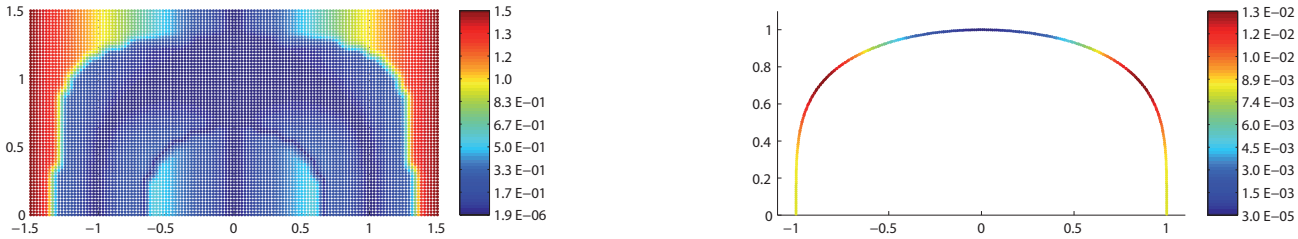

Figure 6.14: error for $h = \frac{1}{16}$ and $n = 4$, *left:* on H^+, *right:* on ω_3

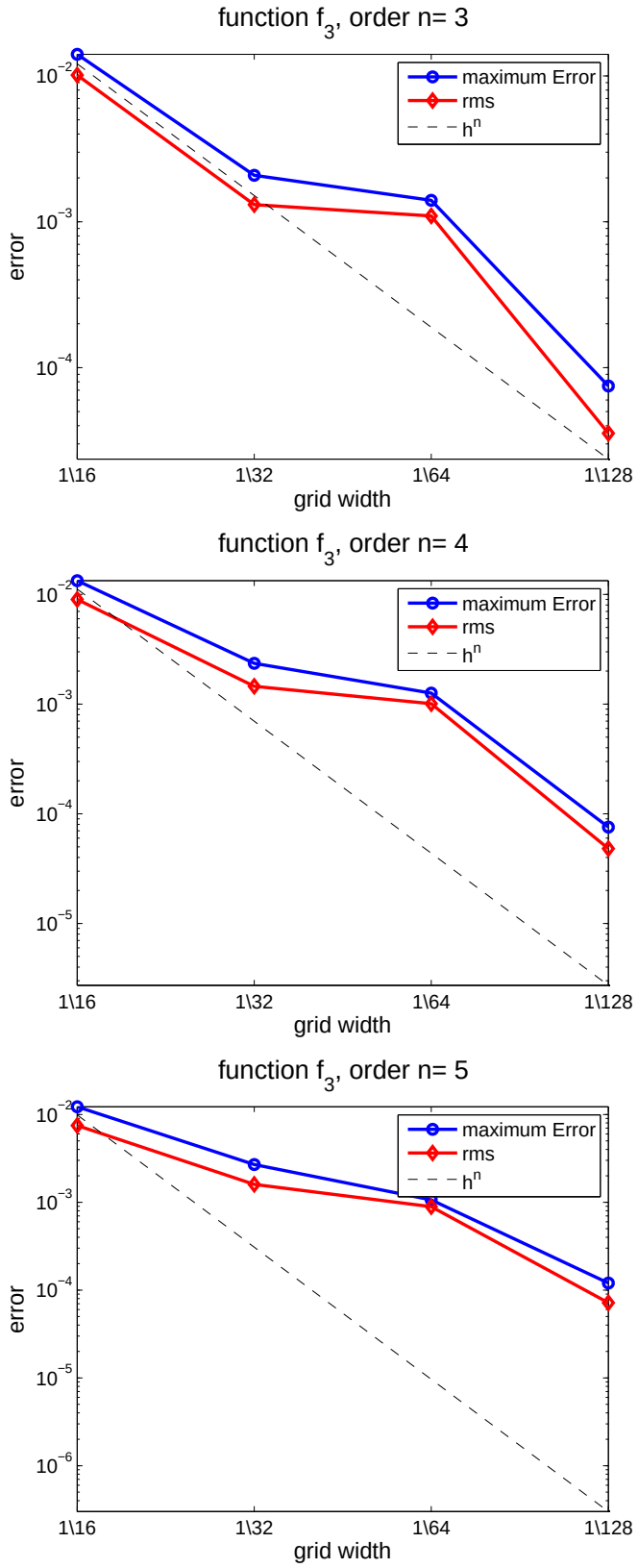

Figure 6.15: experimental rate of convergence on ω_3

7 Conclusion

In the present thesis we introduced three methods. The first method, the Ambient B-spline Method, handles the problem of function approximation on compact submanifolds embedded in $\mathbb{R}^d$. It returns a C^k-approximation of a given function on a submanifold. The second and the third method solve intrinsic PDEs. They provide smooth approximations of solutions of linear elliptic second-order PDEs defined on compact embedded submanifolds with arbitrary geometry. Both problems have lots of applications in different areas.

In all three methods the basic workflow is of same structure: The problem is extended to some ambient domain of the submanifold. There, we solve the problem with already known techniques based on the Cartesian coordinate system. Finally, the result is restricted to the submanifold again. In this way we increase the dimension but not the complexity of the problem. All three methods are novel, straightforward and easy to implement. Neither a complicated functional space has to be constructed, nor do we need a discretisation of the manifold. Not even a parametrisation is needed. All tools are simple by construction and well known. Due to the well elaborated structure of the ambient domain we do not need extended B-splines or even web-splines. We profit from the simplicity of standard tensor product B-splines without having any stability issues. The presented methods are not restricted to manifolds of certain genus or dimension but work for any smooth compact submanifold of codimension one.

The Ambient B-spline Method provides a way to approximate functions or discrete function values on manifolds. We gave some theoretical analysis on the error convergence of the Ambient B-spline Method. As a main result we prove that

$$\|\Delta\|_{W_p^m(\omega)} \leq c\, h^{n-m} \|f\|_{W_p^n(\omega)}, \quad m < n-1$$

for the error Δ, a grid width h, order n and a constant c that is independent of h. The used norm $\|\cdot\|_{W_p^n(\omega)}$ is a Sobolev norm defined on submanifolds. The method shows optimal error behaviour. While the workflow of the method can be applied to submanifolds with arbitrary codimension the presented proof is limited to compact submanifold of codimension one.

A detailed implementation in MATLAB is suggested. The simplicity of the implementation is one of the most beneficial aspects of the Ambient B-spline Method. Numerical results confirm the theoretical results as well as the correctness of the implementation. We apply test functions on submanifolds embedded in $\mathbb{R}^3$. For different orders we find an error convergence in $\mathcal{O}(h^n)$. Not only the convergence but also the absolute value of the error shows good results in the magnitute of 10^{-6} for fine grids.

The Ambient B-spline Method can also be used for higher dimensional functions. In that way we can model surfaces by the Ambient B-spline Method (see [Leh13]). An improvement of the method using hierachical B-splines is also suggested in [Leh13].

A proof of the convergence order for submanifolds of higher codimension is still an open problem. Another aspect that is left for future work is the workflow and theoretical results for non-compact submanifolds or submanifolds with boundary.

The second method presented in this thesis deals with intrinsic linear elliptic second-order PDEs: the Ambient Signed Distance Method. The intrinsic PDE is extended to an embedding PDE on the extended ambient domain of the submanifold. While extending the PDE we make use of the

signed distance function. A simple modification is applied on the differential operators to obtain the embedding PDE.

The embedding PDE is defined on a full-dimensional subset of $\mathbb{R}^d$. Therefore, we need to define boundary conditions. Due to the structure of the domain standard formulas, like Neumann boundary conditions, can not be applied. In this thesis we presented a new formulation of boundary conditions. We thicken the boundary such that it becomes d-dimensional.

We use the Finite Element Method to solve the embedding PDE. Here, standard tensor product B-splines span the finite element space. The method returns a spline of order n. The order and thus the differentiability of the solution can be chosen arbitrarily. Most of the methods presently known construct linear approximations.

An implementation in MATLAB is described and tested on 2-dimensional experiments. These show good results. The error decays with the grid width in $\mathcal{O}(h^n)$. Even the absolute value of the error is very good. One more benefit of the Ambient Signed Distance Method is that ellipticity is preserved. The method of Dzuik and Elliot presented in [DE13] as well as the method suggested by Bertalmio et. al. in [BCOS01] involve a projection. The projection matrix has zero as an eigenvalue which leads to a degenerated problem. Ellipticity is then lost.

The Ambient Signed Distance Method can be generalised to any linear elliptic second-order PDE defined on any embedded smooth compact submanifold. It is independent of the genus or the dimension. Yet, the submanifold should be of codimension one and there should be a signed distance function given. The basic concept of this method can easily be translated into the case of intrinsic linear second-order parabolic PDEs. One can also apply this method on non-compact methods as long as there exists an embedded tubular neighbourhood. Therefore, we cover a huge field of application.

As future work, a theoretical analysis on the error behaviour of the method is left. Some thoughts on this topic were already given in this thesis. Moreover, it is still open if the idea of this method can be applied to higher order PDEs.

The Ambient Level Set Method handles the same problem as the Ambient Signed Distance Method. Here, we do not need the signed distance function but work with any level set function. The Ambient Level Set Method translates the intrinsic PDE into an embedding PDE of a specified shape. To do so, some ordinary differential equations need to be solved in advance. All positive properties of the Ambient Signed Distance Method are adopted, like the simplicity of the used techniques and the implementation as well as stability of the used tensor product B-splines. Numerical tests show that the result of the method converges to the exact solution for shrinking grid width. The convergence order seems to depend on the shape of the manifold. Therefore, it is left as future work to optimise the method. We can try hierarchical B-splines on this method. In that way finer grids can be used where the curvature of the manifold is high. It can also be tested if normalising the flow improves the error behaviour of the method. Finally, generalisation to higher order PDEs and a theoretical proof on the error behaviour are left as future work.

Bibliography

[AD98] Ambrosio, L. and N. Dancer: *Calculus of Variations and Partial Differential Equations*. Springer, 1998.

[BCOS01] Bertalmío, M., L.T. Cheng, S. Osher, and G. Sapiro: *Variational problems and partial differential equations on implicit surfaces*. Journal of Computational Physics, 174:759–780, 2001.

[BH70] Bramble, J.H. and S.R. Hilbert: *Estimation of linear functionals on Sobolev spaces with applicaton to fourier transforms and spline interpolation*. SIAM J. Numer. Anal., 7(1):112–124, 1970.

[Bra97] Braess, D.: *Finite Elemente*. Springer, 1997.

[Bre93] Bredon, G.E.: *Topology and Geometry*, volume 139. Springer, 1993.

[CLB+09] Chuang, M., L. Luo, B.J. Brown, S. Rusinkiewicz, and M. Kazhdan: *Estimating the Laplace Beltrami operator by restricting 3d functions*. Eurographics Symposium on Geometry Processing, 28(5), 2009.

[CS96] Constantine, G.M. and T.H. Savits: *A multivariate Faa di Bruno formula with applications*. Trans. Amer. Math. Soc., 348:503–520, 1996.

[dB78] Boor, C. de: *A Practicle Guide to Splines*. Springer, 1978.

[DDEH10] Deckelnick, K., G. Dzuik, C.M. Elliott, and C. Heine: *An h-narrow band finite element method for elliptic equations on implicit surfaces*. IMA Jounal of Numerical Analysis, 2010.

[DDS80] Dahmen, W., R. DeVore, and K. Scherer: *Multi-dimensional spline approximation*. SIAM J. Numer. Anal., 17(3), june 1980.

[DE13] Dzuik, G. and C.M. Elliott: *Finite element method for surface PDEs*. Acta Numerica, pages 289–396, 2013.

[DPR13] Davydov, O., J. Prasiswa, and U. Reif: *Two-stage approximation methods with extended b-splines*. Mathematics of Computation, 2013.

[Dri03] Driver, B.K.: *Analysis Tools with Applications*. Springer, june 2003.

[EB93] Evans, D.J. and N.M. Bahoshy: *Parallel Romberg interation of multidimensional integrals*. Parallel Algorithms and Applications, 3:271–286, 1993.

[Foo84] Foote, R. L.: *Regularity of the distance function*. Proceedings of the American Mathematical Society, 92:153–153, 1984.

[GHQ06] Gu, X., Y. He, and H. Qin: *Manifold splines*. Graph. Models, 68(3):237–254, 2006.

[Gre03] Greer, J. B.: *An improvement of a recent Eulerian method for solving PDEs on general geometries*. Journal of Scientific Computing, 29(3):321–352, 2003.

[GT01] Gilbarg, D. and N.S. Trudinger: *Elliptic Partial Differential Equations of Second Order*. Springer, 2001.

[HG00] Hubeli, A. and M. Gross: *A survay of surface representations for geometric modeling*. Technical Report 335, ETH Zürich, 2000.

[HH13] Höllig, K. and J. Hörner: *Approximation and Modeling with B-Splines*. Society for Industrial and Applied Mathematics, 2013.

[HJ98] Han, B. and R. Jia: *Multivariate refinement equations and convergence of subdivision schemes*. SIAM J. Math. Anal., 29(5):1177–1199, September 1998.

[Höl03] Höllig, K.: *Finite Element Method with B-Splines*. Society for Industrial and Applied Mathematics, 2003.

[HQ04] He, Y. and H. Qin: *Surface reconstruction with triangular b-splines*. Geometric Modeling and Processing, 2004.

[HRW01] Höllig, K., U. Reif, and J. Wipper: *Weighted extended b-spline approximation of Dirichlet problems*. SIAM J. Numer. Anal., 39(2):442–462, 2001.

[HW05] Hu, S. and J. Wallner: *A second order algorithm for orthogonal projection onto curves and surfaces*. Comp. Aided Geom. Des, 22:251–260, 2005.

[Jos98] Jost, J.: *Partielle Differentialgleichungen*. Springer, 1998.

[Leh13] Lehmann, N.: *Modeling with Ambient B-Splines*. PhD thesis, Technische Universität Darmstadt, 2013.

[LR12] Lehmann, N. and U. Reif: *Notes in the curvature tensor*. Graph. Models, 74(6):321–325, november 2012.

[LWCT07] Lui, L.M., Y. Wang, T.F. Chan, and P.M. Thompson: *Brain anatomical feature detection by solving partial differential equations on general manifolds*. Discrete and Continuous Dynamical Systems - Series B, 7(3):605–618, May 2007.

[Möß06] Mößner, B.: *B-Splines als Finite Elemente*. PhD thesis, Technische Universität Darmstadt, 2006.

[MR08a] Macdonald, C.B. and S.J. Ruuth: *Level set equations on surfaces via the closest point method*. Journal of Scientific Computing, 35(2):219–240, 2008.

[MR08b] Mößner, B. and U. Reif: *Stability of tensor product b-splines on arbitrary domains*. Journal of Approximation Theory, 2008.

[MR09] Macdonald, C.B. and S.J. Ruuth: *The implicit closest point method for numerical solution of partial differential equations on surfaces*. SIAM Journal of Scientific Computing, 31, 2009.

[PP93] Pinkall, U. and K. Polthier: *Computing discrete minimal surfaces and their conjugates*. Experimental Mathematics, 2, 1993.

[Pra09] Prasiswa, J.: *Lokale und Globale Algorithmen zur Approximatio mit Erweiterten B-Splines*. PhD thesis, Technische Universität Darmstadt, 2009.

[Rei07] Reif, U.: *An Appropriate Geometric Invariant for the C^2-Analysis of Subdivision Surfaces*. Springer, 2007.

[Rei12] Reif, U.: *Polynomial approximation on domains bounded by diffeomorphic images of graphs*. Journal of Approximation theory, 164:954–970, 2012.

[RM08] Ruuth, S.J. and B. Merriman: *A simple embedding method for solving partial differential equations on surfaces*. Journal of Computational Physics, 227(3):1943–1961, 2008.

[Sch80] Schumaker, L.: *Spline Functions Basic Theory*. Wiley-Interscience, 1980.

[Sis11] Sissouno, N.: *Multivariate Splineapproximation auf Gebieten*. PhD thesis, Technische Universität Darmstadt, 2011.

[TMR09] Tian, L., C.B. Macdonald, and S.J. Ruuth: *Segmentation on surfaces with the closest point method*. Image Processing (ICIP), 2009.

[vGMM13] Glehn, I. von, T. März, and C.B. Macdonald: *An embedding method-of-lines approach to solving partial differential equations on surfaces*. SIAM Journal on Scientific Computin, 2013.

[War08] Wardetzky, M.: *Convergence of the cotangent formula: an overview*. Discrete Differential Geometry, 2008.

[WHL+07] Wang, H., Y. He, X. Li, X. Gu, and H. Qin: *Polycube splines*. Proceedings of the 2007 CM symposium on Solid and physical modeling, pages 241–251, 2007.

[WMKG07] Wardetzky, M., S. Mathur, F. Kälberer, and E. Grinspun: *Discrete Laplace operators: No free lunch*. Eurographics Symposium on Geometry Processing, 2007.

[YT99] Yngve, G. and G. Turk: *Creating smooth implicit surfaces from polygonal meshes*. Technical report, Graphics, Visualization and Usability Center, 1999.

[ZCMO96] Zhao, H. K., T.F. Chan, B. Merriman, and S. Osher: *A variational level set approach to multiphase motion*. Journal of Computational Physics, 127, 1996.

[ZQ05] Zheng, W. and H. Qin: *On Friedrichs'-Poincaré-type inequalities*. J. Math. Anal. Appl., 304:542–551, 2005.

Wissenschaftlicher Werdegang

Sonja Odathuparambil

02. Januar 1986	geboren in Frankfurt am Main, Deutschland
1992 – 2005	Schulbesuch
Juni 2005	Abitur an der an der Goetheschule Neu-Isenburg, Deutschland
2005 – 2008	Bachelorstudium in *Mathematics with Computer Science* an der Technischen Universität Darmstadt
März 2008	Bachelor of Science
2008 – 2011	Masterstudium in Mathematik mit Nebenfach Informatik an der Technischen Universität Darmstadt
April 2011	Master of Science
seit April 2011	Wissenschaftliche Mitarbeiterin am Fachbereich Mathematik (Arbeitsgruppe Geometrie und Approximation) der Technischen Universität Darmstadt
20. Mai 2016	Einreichung der Dissertation
13. Juli 2016	Tag der mündlichen Prüfung